

IBM InfoSphere Data Replication
Version 10.1.3

*Référence et guide pour la réplication
Q et la publication d'événements*



IBM InfoSphere Data Replication
Version 10.1.3

*Référence et guide pour la réplication
Q et la publication d'événements*



Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques et marques», à la page 603.

Première édition - mai 2012

Réf. US : SC19-3637-00

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
17, avenue de l'Europe
92275 Bois-Colombes Cedex*

© Copyright IBM Corporation 2004, 2012.

Table des matières

Avis aux lecteurs canadiens. ix

Chapitre 1. Conversion de données pour la réplication Q et la publication d'événements 1

Pages de code pour la réplication Q	1
Pages de codes pour la publication d'événements	2

Chapitre 2. Configuration des ID utilisateur et des mots de passe 3

Authentification requise sous Linux, UNIX et Windows	3
Autorisations nécessaires pour la réplication Q et la publication d'événements	3
Autorisations nécessaires pour le programme Q Capture	4
Autorisations nécessaires pour le programme Q Apply	4
Autorisations requises pour le Centre de réplication et le programme ASNCLP	5
Exigences de connectivité pour la réplication Q et la publication d'événements	6
Gestion des ID utilisateur et des mots de passe pour des serveurs distants (Linux, UNIX, Windows)	7

Chapitre 3. Configuration de WebSphere MQ pour la réplication Q et la publication d'événements 9

Objets WebSphere MQ nécessaires à la réplication Q et à la publication d'événements - Présentation Objets	9
Objets WebSphere MQ requis pour la réplication unidirectionnelle (à distance)..	9
Objets WebSphere MQ nécessaires à la réplication unidirectionnelle (même système)	11
Objets WebSphere MQ requis pour la publication d'événements.	13
Objets WebSphere MQ requis pour la réplication bidirectionnelle ou la réplication entre homologues (deux serveurs distants)	14
Objets WebSphere MQ requis pour la réplication entre homologues (trois serveurs distants ou plus)	16
Paramètres requis pour les objets WebSphere MQ	18
Exemples de commandes de création d'objets WebSphere MQ pour la réplication Q et la publication d'événements.	23
Exécution des programmes de réplication sur un client WebSphere MQ	25
Validation des objets WebSphere MQ pour la réplication Q et pour la publication d'événements	27
Contrôles de validation WebSphere MQ effectués par les outils de réplication	29
Exigences en matière de connectivité et d'autorisation pour les objets WebSphere MQ	31

Exigences de stockage pour WebSphere MQ pour la réplication Q et la publication d'événements	33
Taille des messages WebSphere MQ	34
Considérations relatives à la longueur de file d'attente pour les valeurs d'objets LOB	35
Regroupement de gestionnaires de files d'attente dans la réplication Q et la publication d'événements.	36

Chapitre 4. Configuration des bases de données pour la réplication Q et la publication d'événements (Linux, UNIX et Windows) 37

Obligatoire : Définition de DATA CAPTURE CHANGES sur les tables source DB2 et les tables système DB2 for z/OS.	37
Configuration pour les bases de données DB2 distantes ou antérieures	38
Configuration pour GDPS Active/Active (z/OS)	41
Définition des variables d'environnement (Linux, UNIX, Windows)	43
Définition de la variable d'environnement TMPDIR Linux, UNIX).	44
Résolution des problèmes de mémoire pour Q Capture et Q Apply (AIX)	45
Configuration de la base de données source en vue d'une interaction avec le programme Q Capture (Linux, UNIX, Windows)	47
Configuration de la base de données cible en vue d'une interaction avec le programme Q Apply (Linux, UNIX et Windows)	48
Facultatif : liaison des packages de programmes (Linux, UNIX, Windows)	49
Facultatif : liaison des packages du programme Q Capture (Linux, UNIX et Windows)	49
Facultatif : liaison des packages de programme Q Apply (Linux, UNIX et Windows)	50
Facultatif : liaison des packages du moniteur d'alertes de réplication (Linux, UNIX et Windows)	51
Création de tables de contrôle pour les programmes Q Capture et Q Apply.	52
Création de tables de contrôle sur une version différente (Linux, UNIX, Windows)	54
Configuration pour les bases de données DB2 distantes ou antérieures	56

Chapitre 5. Configuration de la réplication Q unidirectionnelle 61

Réplication unidirectionnelle.	61
Regroupement de mappes de file de réplication et d'abonnements Q	62
Création de mappes de files de réplication	64
Création d'abonnements Q pour une réplication unidirectionnelle.	66

Création de profils d'objet cible.	67
Création d'abonnements Q pour une réplication unidirectionnelle.	69
Colonnes source d'abonnements Q (réplication unidirectionnelle)	71
Fréquence d'envoi d'un message (réplication unidirectionnelle) par le programme Q Capture	72
Conditions de recherche pour filtrer des lignes (réplication unidirectionnelle)	73
Variables d'enregistrement de journal pour filtrer des lignes (réplication unidirectionnelle).	76
Comment mapper des colonnes source vers des colonnes cible (réplication unidirectionnelle)	78
Utilisation d'expressions dans la réplication Q.	80
Index ou colonnes de clé des cibles (réplication unidirectionnelle)	87
Options de traitement de conditions imprévues dans la table cible (réplication unidirectionnelle)	89
Options relatives aux erreurs pour la réplication Q.	92

Chapitre 6. Configuration de la réplication Q multidirectionnelle. 97

Réplication bidirectionnelle	97
Réplication entre homologues	99
Réplication bidirectionnelle et réplication entre homologues	105
Création d'abonnements Q pour une réplication bidirectionnelle.	106
Amélioration des performances de la réplication bidirectionnelle grâce à la table IBMQREP_IGNTRAN	107
Options de détection de conflit (réplication bidirectionnelle)	109
Création d'abonnements Q pour une réplication entre homologues	114
Création d'abonnements Q pour une réplication entre homologues avec deux serveurs	114
Création d'abonnements Q pour une réplication entre homologues avec trois serveurs ou plus	116
Démarrage d'une réplication bidirectionnelle ou d'une réplication entre homologues avec deux serveurs	119
Démarrage d'une réplication entre homologues avec trois serveurs ou plus	120
Arrêt d'une réplication bidirectionnelle ou d'une réplication entre homologues avec deux serveurs	122
Arrêt d'une réplication entre homologues avec trois serveurs ou plus	122

Chapitre 7. Réplication des opérations de langage de définition de données (DDL). 125

Abonnements de niveau schéma et réplication DDL	125
Gestion des opérations DDL par Q Capture au niveau de la base de données source	128
Création d'abonnements de niveau schéma	132
Création de profils pour des abonnements Q de niveau table	134

Attribut DATA CAPTURE CHANGES et abonnements de niveau schéma	135
Démarrage d'abonnements de niveau schéma	135
Changement des options de niveau table utilisés par des abonnements de niveau schéma	137
Arrêt des abonnements de niveau schéma.	138
Suppression d'abonnements de niveau schéma	140
Activation d'une réplication automatique des colonnes nouvellement ajoutées à partir de la table source.	140
Ajout de colonnes existantes à un abonnement Q (réplication unidirectionnelle)	141
Ajout de colonnes existantes à un abonnement Q (réplication bidirectionnelle ou entre homologues)	144
Réplication automatique des opérations ALTER TABLE ALTER COLUMN SET DATA TYPE	146

Chapitre 8. La réplication Q dans les cibles fédérées. 149

Paramétrage de la réplication Q vers des cibles fédérées	151
Configuration du serveur fédéré pour la réplication Q	153
Création d'objets fédérés pour la réplication Q	154
Création de tables de contrôle Q Apply pour la réplication Q fédérée	156
Création d'abonnements Q pour la réplication Q fédérée	158

Chapitre 9. Réplication Q à partir de sources Oracle. 161

Configuration de la réplication Q à partir de sources Oracle	162
Support client Oracle pour la réplication Q	164
Configuration de la source Oracle	165
Configuration d'une base de données source Oracle pour travailler avec un programme Q Capture	165
Comment le programme Q Capture travaille avec la fonctionnalité Oracle LogMiner	166
Configuration de la fonctionnalité Oracle LogMiner.	168
Création d'un fichier de configuration ASNCLP	169
Création de tables de contrôle Q Capture pour une source Oracle	170
Création d'abonnements Q pour les sources Oracle	172
Options de chargement pour la réplication Q à partir d'une source Oracle	174
Démarrage d'un programme Q Capture pour des sources Oracle	176
Comment Q Capture gère les modifications des tables source Oracle	177
Types de données Oracle	178

Chapitre 10. Manipulation des données à l'aide des procédures stockées en tant que cibles 181

Procédures stockées pour la manipulation des données source pour la réplication Q	181
---	-----

Écriture de procédures stockées pour manipuler des données source pour la réplication Q	183
Paramètre de procédure stockée qui identifie le type d'opération de la réplication Q	185
Paramètre de procédure stockée qui indique si chaque colonne source a été supprimée.	187
Paramètres de procédure stockée qui identifient la transaction pour la réplication Q	188
Paramètres de procédure stockée qui mappent vers des colonnes source pour la réplication Q	188

Chapitre 11. Chargement des tables cibles pour la réplication Q 191

Recommandations relatives au chargement de tables cible pour une réplication Q	191
Option de chargement automatique pour la réplication Q	192
Utilitaires employés pour l'option de chargement automatique de réplication Q	192
Considérations relatives au chargement automatique pour z/OS	194
Définition de pseudonymes pour l'option de chargement automatique de réplication Q	194
Vérification de la définition des paramètres d'accès simultané appropriés pour les pseudonymes utilisés pour le chargement	195
Chargement manuel d'une table cible	196
Option Aucun chargement pour la réplication Q	199
Options de chargement concernant différents types de réplication Q	199
Réplication d'opérations de chargement de la table source	203
Comment les contraintes sur la table source affectent la réplication des opérations de chargement	205
Chargement de tables cible dans un scénario de consolidation de données	206
Réplication Q et publication d'événements pour plusieurs partitions de base de données	208
Réplication des tables partitionnées DB2 : Version 9.7 groupe de correctifs 1 et antérieurs (Linux, UNIX, Windows)	209
Réplication des tables partitionnées DB2 : Version 9.7 groupe de correctifs 2 et ultérieurs (Linux, UNIX, Windows)	210
Tables CCD dans la réplication Q.	212

Chapitre 12. Création de publications 219

Regroupement des publications et des mappes de files d'attente pour la publication	219
Création de mappes de files pour la publication	220
Création de publications.	223
Création de publications.	223
Colonnes source pour les publications XML	226
Publication par le programme Q Capture d'un message pour les publications	226
Conditions de recherche pour filtrer les lignes dans les publications XML	227
Colonnes de clé pour les publications XML	230

Options pour l'intégration de colonnes non modifiées dans les messages pour les publications XML	230
Options pour l'intégration de valeurs avant dans les messages pour les publications	231

Chapitre 13. Considérations relatives au type de données 233

Restrictions générales de données pour la réplication Q et la publication d'événements	233
Considérations concernant les types de données d'objet LOB pour la réplication Q et la publication d'événements	235
type de données XML	237
Réplication entre colonnes XML et LOB	238
Réplication de nouveaux types de données DB2 version 9.7 (Linux, UNIX, Windows)	238
Réplication de tables avec des colonnes d'identité	240

Chapitre 14. Utilisation des commandes et des scripts générés par les outils d'administration de la réplication. 243

Exécution et enregistrement des scripts générés par les outils d'administration de la réplication	243
Exécuter et enregistrer les commandes (Centre de réplication)	245

Chapitre 15. Utilisation d'un programme Q Capture 247

Démarrage d'un programme Q Capture	247
Démarrage de Q Capture à partir d'un emplacement connu dans le journal DB2	249
Indiquer à Q Capture des points de redémarrage pour des files d'attente d'envoi individuelles ou des partitions individuelles de données (z/OS)	252
Considérations relatives à l'utilisation du mode démarrage à froid	254
Changement des paramètres de Q Capture	255
Méthodes de changement des paramètres d'exploitation de Q Capture	255
Modification de paramètres pendant l'exécution d'un programme Q Capture	257
Changement des paramètres Q Capture sauvegardés dans la table IBMQREP_CAPPARMS	258
Indication des transactions indésirables à ignorer par le programme Q Capture	259
Arrêt d'un programme Q Capture	263
Arrêt d'un programme Q Capture à un point spécifié	264
Démarrage des abonnements Q	268
Arrêt des abonnements Q	270
Démarrage des publications	271
Arrêt des publications	272
Gestion de l'activité des messages Q Capture au niveau de la file d'attente d'envoi.	273
Démarrage de l'activité de message sur une ou plusieurs files d'attente d'envoi	276

Arrêt de l'activité de message sur une ou plusieurs files d'attente d'envoi	278
---	-----

Chapitre 16. Exécution d'un programme Q Apply 281

Démarrage d'un programme Q Apply	281
Changement des paramètres de Q Apply	283
Modification de paramètres pendant l'exécution d'un programme Q Apply	283
Changement des paramètres enregistrés de Q Apply dans la table IBMQREP_APPLYPARMS	284
Arrêt d'un programme Q Apply	285
Arrêt du traitement des messages dans une file d'attente de réception.	286
Démarrage du traitement de messages dans une file d'attente de réception	288
Indication des transactions à ignorer par un programme Q Apply	289

Chapitre 17. Modification de l'environnement de réplication Q . . . 293

Modification des propriétés des abonnements Q unidirectionnels	293
Ajout de colonnes existantes à un abonnement Q (réplication unidirectionnelle)	295
Ajout de colonnes existantes à un abonnement Q (réplication bidirectionnelle ou entre homologues)	297
Gestion des opérations DDL par Q Capture au niveau de la base de données source	300
Modification des propriétés des mappes de files de réplication	304
Modification des noms de file d'attente lorsque la mappe de files d'attente est utilisée par les abonnements Q.	306
Redémarrage des abonnements Q échoués sans les supprimer ni les recréer	307
Placer des abonnements Q dans un spill mode à des fins de maintenance.	308
Suppression d'abonnements Q.	309
Suppression des mappes de files de réplication	310
Suppression des tables de contrôle Q Capture ou Q Apply	311

Chapitre 18. Modification d'un environnement de publication d'événements 313

Modification des propriétés de publications	313
Ajout de colonnes aux publications existantes	315
Suppression de publications	316
Modification des propriétés des mappes de files pour la publication	317
Suppression des mappes de files de réplication	318
Suppression des tables de contrôle Q Capture	319

Chapitre 19. Vérification du statut des programmes de réplication Q et de publication d'événements 321

Vérification de l'état des programmes de réplication Q et de publication d'événements	321
---	-----

Unités d'exécution des programmes Q Capture, Q Apply et du moniteur d'alertes de réplication	323
Temps d'attente.	326
Exceptions	327

Chapitre 20. Gestion d'un environnement de réplication Q et de publication d'événements 331

Considérations relatives à la gestion des systèmes source de réplication Q et de publication d'événements	331
Gestion des tables source dans un environnement de réplication Q et de publication d'événements	331
Conservation de fichiers journaux pour la réplication Q et la publication d'événements	331
Considérations pour la gestion des dictionnaires de compression dans la réplication Q et la publication d'événements (z/OS)	334
Gestion des tables de contrôle dans la réplication Q et la publication d'événements.	335
Elagage des tables de contrôle dans la réplication Q et la publication d'événements	335
Considérations relatives à l'utilisation de l'utilitaire RUNSTATS sur les tables de contrôle dans un environnement de réplication Q et de publication d'événements	337
Réorganisation des tables de contrôle	337
Quand les programmes de réplication ne peuvent pas se connecter à leur serveur DB2	338
Gestion des tables cible	338
Remarques relatives à la redéfinition de modules et de plans pour la réplication Q et la publication d'événements	340

Chapitre 21. Comparaison et réparation de tables 341

Utilitaire de comparaison de tables (asntdiff)	341
Exécution de l'utilitaire asntdiff en mode parallèle (sous z/OS)	347
Utilitaire de réparation des tables (asntrep)	350
Gestion des fonctions de compatibilité SQL DB2 par l'utilitaire de comparaison.	350

Chapitre 22. Utilisation des services système pour exécuter des programmes de réplication Q et de publication d'événements 355

Utilisation des services système z/OS pour exécuter les programmes de réplication Q et de publication d'événements	355
Exécution des programmes de réplication Q et de publication d'événements à l'aide de JCL	355
Exécution des programmes de réplication Q et de publication d'événements avec les tâches démarrées par le système	361

Utilisation du gestionnaire ARM (Automatic Restart Manager) pour redémarrer automatiquement la réplication et la publication (z/OS)	362
Services de réplication (Windows)	363
Description des services Windows pour la réplication	363
Création d'un service de réplication	364
Lancement d'un service de réplication	365
Arrêt d'un service de réplication	365
Affichage d'une liste des services de réplication	366
Suppression d'un service de réplication	366
Planification des programmes de réplication	366
Planification des programmes de publication d'événements et de réplication (Linux, UNIX)	366
Planification des programmes de réplication (Windows)	367
Planification des programmes de réplication et de publication d'événements (z/OS)	367

Chapitre 23. Conventions et instructions de dénomination pour la réplication Q et la publication d'événements - Présentation 369

Conventions de dénomination pour les objets de réplication Q et de publication d'événements	369
Gestion des noms d'objet en minuscules pour la réplication Q et la publication	372

Chapitre 24. Commandes système de réplication Q et de publication d'événements 373

asnqcap : démarrage d'un programme Q Capture	374
Descriptions des paramètres asnqcap	375
Exemples d'utilisation de la commande asnqcap	397
asnoqcap : Démarrage d'un programme Q Capture pour une base de données Oracle	398
asnqcmd : utilisation d'un programme Q Capture en cours d'exécution	407
asnoqcmd : Gestion d'un programme Q Capture en cours d'exécution sur des bases de données Oracle	412
asnqapp : lancement d'un programme Q Apply	416
Descriptions des paramètres asnqapp	417
Exemples d'utilisation de la commande asnqapp	435
asnqcmd : utilisation d'un programme Q Apply en cours d'exécution	436
asnpwd : création et gestion des fichiers de mots de passe	441
anscrt : création d'un service de réplication	445
ansndrop : suppression d'un service de réplication	448
asnslst : liste des services de réplication	449
asntdiff : comparaison des données dans les tables source et cible (Linux, UNIX, Windows)	450
asntdiff : comparaison des données dans les tables source et cible (z/OS)	454
Option de commande asntdiff -f (fichier en entrée)	461
asntrc : Utilisation de la fonction trace de réplication	465

asntrep : réparation des différences entre les tables source et cible	473
asnqanalyze : fonctionnement de l'analyseur de réplication Q (Replication Analyzer)	475
asnqmfmt : mise en forme et visualisation des messages de réplication Q et de publication	484
asnqxfmt : formatage et affichage des messages de publication d'événements (z/OS)	486

Chapitre 25. Tables de contrôle pour la réplication Q et la publication d'événements 489

Tables de contrôle sur le serveur Q Capture	489
Table IBMQREP_ADMINMSG	490
Table IBMQREP_CAPENQ	491
Table IBMQREP_CAPENVINFO	492
Table IBMQREP_CAPMON	492
Table IBMQREP_CAPPARMS	494
Table IBMQREP_CAPQMON	500
Table IBMQREP_CAPTRACE	502
Table IBMQREP_COLVERSION	503
Table IBMQREP_EOLFLUSH	504
Table IBMQREP_EXCLSCHEMA	505
Table IBMQREP_IGNTRAN	506
Table IBMQREP_IGNTRANTRC	506
Table IBMQREP_PART_HIST (Linux, UNIX, Windows)	507
Table IBMQREP_SCHEMASUBS	509
Table IBMQREP_SENDQUEUES	511
Table IBMQREP_SIGNAL	515
Table IBMQREP_SRC_COLS	520
Table IBMQREP_SRCH_COND	521
Table IBMQREP_SUBS	522
Table IBMQREP_SUBS_PROF	528
Table IBMQREP_TABVERSION	529
Tables de contrôle sur le serveur Q Apply	530
Table IBMQREP_APPLYENQ	531
Table IBMQREP_APPENVINFO	532
Table IBMQREP_APPLYMON	533
Table IBMQREP_APPLYPARMS	538
Table IBMQREP_APPLYTRACE	545
Table IBMQREP_DELTOMB	546
Table IBMQREP_DONEMSG	547
Table IBMQREP_EXCEPTIONS, table	547
Table IBMQREP_RECVQUEUES	550
Table IBMQREP_SAVERI	552
Table IBMQREP_SPILLQS	554
Table IBMQREP_SPILLEDROW	554
Table IBMQREP_TARGETS	555
Table IBMQREP_TRG_COLS	563

Structure détaillée des colonnes de gestion des versions pour la réplication entre homologues	564
---	-----

Chapitre 26. Structure des messages XML pour la publication d'événements 567

Types de message XML et conditions requises	567
Types de message	567
Conditions techniques requises pour les messages XML	567

Traitement des délimiteurs XML dans les données de type caractères	568
Structure des messages provenant de Q Capture vers une application utilisateur	568
Liste des messages provenant de Q Capture vers une application utilisateur	568
msg : Elément principal des messages XML provenant du programme Q Capture vers une application utilisateur	569
Message de transaction	571
Message d'opération de ligne	580
Message d'objet LOB	582
Message d'abonnement désactivé	584
Message de chargement terminé reçu	585
Message de rapport d'erreurs	586
Message de pulsation.	587
Message de schéma d'abonnement (subSchema)	588
Message d'ajout de colonne.	592
Structure des messages provenant d'une application utilisateur vers Q Capture	593

Liste des messages provenant d'une application utilisateur vers Q Capture	593
msg : Elément principal des messages XML provenant d'une application utilisateur vers Q Capture	594
Message invalidateSendQ	595
Message de chargement terminé	595
Message d'activation d'abonnement	596
Message de désactivation d'abonnement	596
Contacteur IBM	599
Lecture des diagrammes de syntaxe	601
Remarques et marques	603
Marques	607
Index	609

Avis aux lecteurs canadiens

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

Illustrations

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

Terminologie

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

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

Claviers

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








OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

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

Nomenclature

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

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

Brevets

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

Assistance téléphonique

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

Chapitre 1. Conversion de données pour la réplication Q et la publication d'événements

Dans la réplication Q et la publication d'événements, les données sont transmises entre divers serveurs et programmes et nécessitent parfois d'être converties dans des pages de codes différentes.

Pages de code pour la réplication Q

Dans la réplication Q, les données sont transmises entre plusieurs serveurs et programmes, et elles doivent parfois être converties. Les programmes peuvent par exemple avoir différentes pages de codes, ou bien la plateforme ou le processeur peut gérer différemment des valeurs numériques.

Les données sont automatiquement converties au niveau des colonnes, selon les besoins, même si le serveur source et le serveur cible utilisent des pages de codes différentes. Le format endian et les représentations à virgule flottante sont gérés.

Recommandation : Dans la mesure du possible, évitez de convertir les données en utilisant les pages de codes correspondantes pour les programmes ou les serveurs suivants :

- Programme Q Capture
- Programme Q Apply
- Serveur source
- Serveur cible

Si le serveur source et le serveur cible doivent utiliser des pages de codes différentes, utilisez alors les pages de codes correspondantes pour le programme Q Capture et le serveur source et faites de même pour le programme Q Apply et le serveur cible.

Lorsque les données de la table source sont répliquées vers la table cible ou une procédure stockée, le programme Q Capture envoie les données à WebSphere MQ dans le format de la table source. Si nécessaire, les données source sont converties au format de la cible.

Linux UNIX Windows Sous Linux, UNIX et Windows, lorsque le programme Q Apply reçoit les données source de la file d'attente de réception, il convertit chaque colonne de la transaction et toutes les autres données du message dans sa propre page de codes. Le serveur cible s'attend à recevoir des données dans la même page de codes et la même représentation à virgule flottante que le programme Q Apply.

z/OS Sous z/OS, le programme Q Apply ne convertit pas les données source vers sa propre page de codes. A la place, pour chaque opération, le programme Q Apply balise la page de codes source des données de DB2. DB2 convertit ensuite directement la page de codes source en page de codes cible.

Si la table source est définie en EBCDIC ou ASCII et que la table cible est définie en UTF-8, des erreurs SQL peuvent se produire si la colonne cible est trop petite pour gérer l'accroissement du nombre d'octets résultant de la conversion des

données EBCDIC ou ASCII en UTF-8. Pour éviter cette erreur, la colonne de la table cible doit être définie de manière à avoir 3 octets pour chaque caractère stocké à la source. Par exemple, si la colonne source est définie comme CHAR(x), la colonne cible doit être définie comme CHAR(3x).

z/OS **Recommandation** : zSeries représente les valeurs à virgule flottante différemment des unités centrales basées sur Intel, ce qui peut provoquer des pertes de données. Evitez d'utiliser une colonne flottante en tant que clé.

Restriction : Les pages de codes pour les programmes Q Capture et Q Apply ne peuvent pas être en UTF-16.

Si vous projetez de répliquer des données entre des serveurs avec des pages de codes différentes, reportez-vous au *Guide d'administration IBM DB2 : planification* pour déterminer si les pages de codes que vous utilisez sont compatibles.

Pages de codes pour la publication d'événements

Dans la publication d'événements, les données sont converties de la page de codes du serveur source en message XML au format UTF-8 (page de codes 1208, qui est un algorithme de codage XML standard) ou en message délimité dans une page de codes spécifiée par l'utilisateur (page de codes par défaut 1208).

Lorsque l'application utilisateur envoie un message d'administration (par exemple, un message de désactivation d'abonnement) au programme Q Capture au format XML, l'analyseur syntaxique convertit le message à la page de codes du programme Q Capture.

Chapitre 2. Configuration des ID utilisateur et des mots de passe

Pour utiliser les programmes de réplication Q et de publication d'événements, vous devez configurer les ID utilisateur et les mots de passe pour accéder aux serveurs DB2 sur des systèmes distribués.

Authentification requise sous Linux, UNIX et Windows

La réplication Q ne nécessite pas que vous utilisiez un type d'authentification spécifique. Vous pouvez utiliser le type d'authentification compatible qui correspond le mieux aux besoins de votre activité.

La liste suivante offre plus de précisions :

Sources et cibles DB2

- La réplication Q est un programme d'application de base de données.
- Les fonctions client-serveur DB2 sous-jacentes sont aussi transparentes pour la réplication Q qu'elles le sont pour toute application de base de données.
- La réplication Q se connecte aux bases de données DB2 à l'aide de la combinaison classique d'ID et de mot de passe. Vous pouvez utiliser n'importe quel type d'authentification compatible.
- La réplication Q n'a pas d'autre exigence que cela pour le type d'authentification.

Sources Oracle

- Q Capture pour les bases de données Oracle est un programme d'application de base de données Oracle C++ Call Interface (OCCI).
- Q Capture utilise les données d'identification de l'utilisateur Oracle habituelles, qui sont nécessaires pour se connecter à une base de données Oracle. Les données d'identification de l'utilisateur du système d'exploitation ne sont pas utilisées lors de la connexion.
- La valeur du paramètre **capture_server** ou **apply_server** correspond à un alias défini dans un fichier tnsnames.ora. La valeur du paramètre **capture_schema** ou **apply_schema** correspond à l'ID utilisateur fourni au moment de la connexion.
- Vous devez créer un fichier de mot de passe avec l'utilitaire asnpwd. Lorsque Q Capture pour Oracle se connecte à la base de données Oracle, il extrait le mot de passe de ce fichier de mot de passe.

Autorisations nécessaires pour la réplication Q et la publication d'événements

Les ID utilisateur exécutant les programmes de réplication Q et la publication d'événements doivent avoir l'autorisation de se connecter aux serveurs, d'accéder aux tables et de les mettre à jour et d'exécuter d'autres opérations.

Autorisations nécessaires pour le programme Q Capture

Tous les ID utilisateur qui exécutent un programme Q Capture doivent posséder une autorisation pour accéder au catalogue système DB2, accéder à toutes les tables de contrôle Q Capture et les mettre à jour, lire le journal DB2 et exécuter les modules du programme Q Capture.

z/OS Pour obtenir une liste des autorisations nécessaires sous z/OS, voir Autorisations DB2 pour les programmes Q Capture et Capture.

Linux UNIX Windows La liste suivante résume les exigences DB2 et les exigences du système d'exploitation pour Linux, UNIX et Windows :

Conditions requises

Les ID utilisateur exécutant un programme Q Capture doivent posséder les autorisations et les privilèges suivants :

- Droits DBADM ou SYSADM.
- Privilège WRITE sur le répertoire spécifié par le paramètre **capture_path**. Le programme Q Capture crée des fichiers de diagnostic dans ce répertoire.
- **Windows** Droits d'accès pour la création d'objets globaux.

Dans un environnement de base de données partitionnée, les ID utilisateur doivent être capables de se connecter aux partitions de base de données et de lire le fichier des mots de passe.

Autorisations nécessaires pour le programme Q Apply

Tous les ID utilisateur qui exécutent un programme Q Apply doivent posséder une autorisation pour accéder au catalogue système DB2, accéder aux cibles et les mettre à jour, accéder aux tables de contrôle Q et les mettre à jour, exécuter les modules du programme Q Apply et lire le fichier des mots de passe de Q Apply.

z/OS Pour obtenir une liste des autorisations nécessaires sous z/OS, voir Autorisations DB2 pour les programmes Q Apply et Apply.

Linux UNIX Windows

La liste suivante résume les exigences DB2 et les exigences du système d'exploitation pour Linux, UNIX et Windows, ainsi que pour les cibles non DB2.

Conditions requises

Les ID utilisateur exécutant un programme Q Apply doivent posséder les autorisations et les privilèges suivants :

- Droits DBADM ou SYSADM.
- Privilège SELECT pour les tables source si le programme Q Apply est utilisé pour charger des tables cible.
- Privilège WRITE sur le répertoire spécifié par le paramètre **apply_path**. Le programme Q Apply crée des fichiers de diagnostic dans ce répertoire.
- **Windows** Droits d'accès pour la création d'objets globaux.

Si le programme Q Apply utilise l'option LOAD from CURSOR de l'utilitaire LOAD pour charger les tables cible, le serveur Q Apply doit être un serveur fédéré et vous devez créer des pseudonymes, des définitions de serveur et des mappages utilisateur sur le serveur Q Apply. L'ID utilisateur

fourni dans les mappages utilisateur doit posséder le privilège de lecture depuis les pseudonymes sur le serveur fédéré de Q Apply et de lecture depuis les tables source.

Exigences pour les cibles non DB2

Les ID utilisateur exécutant un programme Q Apply doivent posséder les autorisations et les privilèges suivants :


- CREATE TABLE et CREATE INDEX sur la base de données éloignée.
- Le privilège WRITE sur les pseudonymes dans la base de données fédérée et, au travers des mappages utilisateur, le privilège WRITE sur la cible non DB2.

Autorisations requises pour le Centre de réplication et le programme ASNCLP

Les ID utilisateur qui administrent la réplication Q et la publication d'événements avec le Centre de réplication ou le programme de ligne de commande ASNCLP requièrent certaines autorisations sur les bases de données sur les serveurs source et cible.

Vous devez avoir au moins un ID utilisateur sur toutes les bases de données impliquées dans la configuration de réplication et cet ID utilisateur doit avoir les autorisations nécessaires pour effectuer un certain nombre de tâches administratives sur le serveur Q Capture, le serveur Q Apply et le serveur de contrôle du moniteur si vous utilisez le moniteur d'alertes de réplication.

Les autorisations et privilèges suivants sont requis :

- Privilège CONNECT pour le serveur Q Capture, le serveur Q Apply et le serveur de contrôle du moniteur
- Tous les privilèges de tables, d'espaces de table et d'index nécessaires pour créer les tables de contrôle sur le serveur Q Capture, le serveur Q Apply et le serveur de contrôle du moniteur
- Tous les privilèges de tables, d'espaces de table et d'index nécessaires pour créer des cibles sur le serveur Q Apply
- Les privilèges SELECT, UPDATE, INSERT et DELETE pour toutes les tables de contrôle sur le serveur Q Capture, le serveur Q Apply et le serveur de contrôle du moniteur
-  Privilège SELECT pour les tables de catalogue système DB2 for z/OS suivantes :
 - SYSIBM.SYSCHECKS
 - SYSIBM.SYSCOLUMNS
 - SYSIBM.SYSDATABASE
 - SYSIBM.SYSDUMMY1
 - SYSIBM.SYSINDEXES
 - SYSIBM.SYSINDEXPART
 - SYSIBM.SYSKEYCOLUSE
 - SYSIBM.SYSKEYS
 - SYSIBM.SYSKEYTARGETS
 - SYSIBM.SYSPARMS
 - SYSIBM.SYSRELS
 - SYSIBM.SYSROUTINES

- SYSIBM.SYSSTOGROUP
- SYSIBM.SYSTABCONST
- SYSIBM.SYSTABLEPART
- SYSIBM.SYSTABLES
- SYSIBM.SYSTABLESPACE
- SYSIBM.SYSTRIGGERS
- Privilèges pour définir des plans sur chaque base de données DB2 impliquée dans la réplication ou la publication, y compris sur le serveur Q Capture, le serveur Q Apply et le serveur de contrôle du moniteur
- Les privilèges de procédure stockée suivants :

z/OS

Droit EXECUTE sur les procédures stockées suivantes :

- SYSIBM.SQLPROCEDURECOLS
- SYSPROC.DSNWZP
- SYSPROC.ADMIN_INFO_SSID

Linux UNIX Windows

Privilèges pour créer les procédures stockées à l'aide d'une bibliothèque partagée et appeler des procédures stockées.

Pour simplifier l'administration avec le Centre de réplication, vous pouvez utiliser la fenêtre Gérer les mots de passe et la connectivité pour stocker les ID utilisateur pour les serveurs ou les systèmes, ainsi que pour modifier les ID stockés et tester les connexions. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le dossier **Centre de réplication** et sélectionnez **Gérer les mots de passe du Centre de réplication**.

Exigences de connectivité pour la réplication Q et la publication d'événements

Pour répliquer ou publier des données dans un environnement réparti, vous devez installer et configurer la connectivité. Le plus souvent, vous devez également être en mesure de vous connecter à des bases de données ou à des sous-systèmes DB2 distants pour utiliser le programme de ligne de commande ASNCLP, le Centre de réplication ou le moniteur d'alertes de réplication, pour charger des tables cible ou pour insérer des signaux afin d'activer ou de désactiver des abonnements Q ou des publications.

Les exigences de connectivité pour les bases de données ou les sous-systèmes DB2 diffèrent selon votre environnement de réplication ou de publication :

- Le programme ASNCLP ou le Centre de réplication doit être capable de procéder aux connexions suivantes :
 - Vers le serveur Q Capture pour administrer la publication d'événements.
 - Vers le serveur Q Capture et le serveur Q Apply pour administrer la réplication Q.
 - Vers le serveur de contrôle du moniteur pour installer le moniteur d'alertes de réplication.
- Si vous voulez que le programme Q Apply charge automatiquement les cibles avec les données source en utilisant l'utilitaire EXPORT, le programme Q Apply doit pouvoir se connecter au serveur Q Capture. Cette connexion nécessite un fichier de mots de passe créé à l'aide de la commande **asnpwd**.

- Le programme Q Capture doit être en mesure de se connecter à des bases de données partitionnées. Cette connexion nécessite un fichier de mots de passe créé à l'aide de la commande **asnpwd**.
- Si vous utilisez les outils d'administration, les commandes système ou le SQL pour administrer la réplication à partir d'un poste de travail distant, ce dernier doit être en mesure de se connecter au serveur Q Capture, au serveur Q Apply ou au serveur de contrôle du moniteur.

Gestion des ID utilisateur et des mots de passe pour des serveurs distants (Linux, UNIX, Windows)

Dans certains cas, la réplication et la publication d'événements nécessitent un fichier de mots de passe pour stocker les ID utilisateur et les mots de passe afin de se connecter à des serveurs distants.

A propos de cette tâche

Un fichier de mots de passe est requis dans les cas suivants :

- Le programme Apply nécessite un fichier de mots de passe pour accéder à des données sur les serveurs distants (le programme Capture ne nécessite pas de fichier de mots de passe).
- Le programme Q Apply nécessite un fichier de mots de passe pour se connecter au serveur Q Capture et pour que les abonnements Q utilisant l'utilitaire EXPORT chargent les cibles.
- Le programme Q Capture exige un fichier de mots de passe pour se connecter aux bases de données à partitions multiples.
- Si le programme Q Capture s'exécute à distance à partir de la base de données source ou si le programme Q Apply s'exécute à distance à partir de la base de données cible, le programme exige que les fichiers de mots de passe se connectent à la base de données distante.
- Les commandes **asntdiff** et **asntrep** nécessitent des fichiers de mots de passe pour se connecter aux bases de données où les utilitaires comparent ou réparent les différences de table.
- Le moniteur d'alertes de réplication nécessite un fichier de mots de passe pour se connecter à un serveur Q Capture, Capture, Q Apply, ou Apply que vous souhaitez surveiller.

Remarque importante concernant la compatibilité des fichiers de mots de passe : A partir de la version 9.5 avec le groupe de correctifs 2, les fichiers de mots de passe qui sont créés par la commande **asnpwd** utilisent une nouvelle méthode de chiffrement et ne peuvent pas être lus par des versions plus anciennes des programmes et des utilitaires de réplication. Si vous partagez un fichier de mots de passe parmi des programmes et des utilitaires au niveau mixte avec des groupes de correctifs plus anciens, ne recréez pas le fichier de mots de passe à l'aide d'une commande **asnpwd** faisant partie de ces groupes de correctifs ou de groupes plus récents. Les programmes et utilitaires de réplication de ces groupes de correctifs ou de groupes plus récents peuvent toujours fonctionner avec des fichiers de mots de passe plus anciens. De plus, il est impossible de modifier un fichier de mots de passe plus ancien pour utiliser la nouvelle méthode de chiffrement. Vous devez créer un nouveau fichier de mots de passe.

Généralement, la réplication et la publication d'événements prennent en charge les scénarios suivants :

- Création d'un fichier de mots de passe avec une version et utilisation de ce fichier avec une version plus récente. Par exemple, vous pouvez créer un fichier de mots de passe sous V8.2 et l'utiliser avec V9.1 et V9.5.
- Création d'un fichier de mots de passe avec un groupe de correctifs et utilisation de ce fichier avec un groupe de correctifs plus récent au sein de la même version. Par exemple, vous pouvez créer un fichier de mots de passe avec V9.1, groupe de correctifs 3 et l'utiliser avec V9.1, groupe de correctifs 5.
- Création d'un fichier de mots de passe sur un système et utilisation de ce fichier sur un autre système à condition que les critères suivants soient remplis :
 - Les systèmes utilisent la même page de codes.
 - Les systèmes sont tous 32 bits ou tous 64 bits.

Les fichiers de mots de passe chiffrés ne sont pas pris en charge pour Windows x64 jusqu'à la version 9.5, groupe de correctifs 2 ou supérieur.

Procédure

Pour gérer les ID utilisateur et les mots de passe pour des serveurs distants, procédez comme suit :

- Créez un fichier de mots de passe chiffré pour les programmes de réplication et de publication d'événements exécutés sous Linux, UNIX et Windows en utilisant la commande **asnpwd**. Le fichier de mots de passe doit être stocké dans le chemin défini par les paramètres suivants :

Tableau 1. Exigences du fichier de mots de passe

Programme	Paramètre
Apply	apply_path
Q Apply	apply_path
Q Capture	capture_path
moniteur d'alertes de réplication	monitor_path
Commande asntdiff ou asntrep	DIFF_PATH

- Si le programme Q Apply et le moniteur d'alertes de réplication sont exécutés sur le même système, ils peuvent partager le même fichier de mots de passe. Si vous voulez que les programmes partagent un fichier de mots de passe, définissez le même chemin et le même nom de fichier pour les programmes ou utilisez un lien symbolique de partage du même fichier de mots de passe dans les différents répertoires.
- Le Centre de réplication n'utilise pas le fichier de mots de passe créé avec la commande **asnpwd** pour la connexion aux serveurs distants. La première fois que le Centre de réplication doit accéder à une base de données ou à un sous-système, un ID utilisateur et un mot de passe vous sont demandés, qui seront stockés pour un usage ultérieur. Vous pouvez utiliser la fenêtre Gérer les mots de passe et la connectivité pour stocker les ID utilisateur pour les serveurs ou les systèmes, ainsi que pour modifier les ID stockés et tester les connexions. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le dossier **Centre de réplication** et sélectionnez **Gérer les mots de passe du Centre de réplication**.

Chapitre 3. Configuration de WebSphere MQ pour la réplication Q et la publication d'événements

La réplication Q et la publication d'événements utilisent WebSphere MQ (anciennement MQ Series) pour transmettre des données transactionnelles et échanger d'autres messages. La

Recommandation : Créez les gestionnaires de file d'attente, les files d'attente et autres objets pour l'infrastructure de messagerie, avant de créer des objets de réplication tels que les tables de contrôle et les abonnements Q, et avant de démarrer les programmes Q Capture et Q Apply. Les outils et exemples suivants sont disponibles pour vous aider :

- Les générateurs de script MQ que vous pouvez utiliser en cliquant sur l'icône du générateur de script dans la barre de menu du Centre de réplication
- La commande CREATE MQ SCRIPT du programme de ligne de commande ASNCLP
- **z/OS** L'exemple de travail ASNQDEFQ dans l'ensemble de données l'ensemble de données SASNSAMP
- **Linux UNIX Windows** L'exemple de script asnqdefq dans le répertoire SQLLIB/samples/repl/q

Pour plus d'informations sur WebSphere MQ, consultez le centre de documentation WebSphere MQ à l'adresse <http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp>.

Objets WebSphere MQ nécessaires à la réplication Q et à la publication d'événements - Présentation Objets

Selon le type de réplication ou de publication que vous envisagez d'effectuer, vous devrez utiliser divers objets WebSphere MQ.

Pour plus d'informations sur la création d'objets WebSphere MQ, consultez le centre de documentation WebSphere MQ à l'adresse <http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp>.

Objets WebSphere MQ requis pour la réplication unidirectionnelle (à distance).

La réplication Q ou la publication d'événements unidirectionnelle entre serveurs distants nécessite un gestionnaire de files d'attente et des files d'attente pour le programme Q Capture et pour le programme Q Apply.

Les serveurs étant distribués, vous aurez également besoin de files d'attente de transmission et de canaux pour la transmission des transactions et pour la communication au sein d'un réseau.

Les listes qui suivent énumèrent les objets requis pour une réplication unidirectionnelle entre deux serveurs distants :

Objets non canaux du système source

- Un gestionnaire de files d'attente

- Une définition de file d'attente éloignée à utiliser comme file d'attente d'envoi (cette file d'attente pointe vers une file d'attente de réception sur le système cible)
- Une file d'attente locale à utiliser comme file d'attente d'administration
- Une file d'attente locale à utiliser comme file d'attente de redémarrage

Objets non canaux du système cible

- Un gestionnaire de files d'attente
- Une file d'attente locale à utiliser comme file d'attente de réception
- Une définition de file d'attente éloignée à utiliser comme file d'attente d'administration (cette file d'attente pointe vers une file d'attente d'administration sur le système source)
- Une définition de file d'attente modèle pour les files d'attente de déversement locales temporaires créée par le programme Q Apply et utilisée pendant le chargement des tables cible

Remarque : Si vous avez plusieurs programmes Q Apply qui partagent le même gestionnaire de files d'attente, chaque programme Q Apply doit avoir sa propre file d'attente modèle avec un nom qui est unique sous ce gestionnaire de files d'attente.

Un canal de la source à la cible

- Canal émetteur défini au sein du gestionnaire de files d'attente source
- Une file d'attente de transmission locale associée
- Un canal récepteur de mise en correspondance défini au sein du gestionnaire de files d'attente cible

Un canal de la cible à la source

- Un canal émetteur défini au sein du gestionnaire de files d'attente cible
- Une file d'attente de transmission locale associée
- Un canal récepteur de mise en correspondance défini au sein du gestionnaire de files d'attente source

La figure 1, à la page 11 répertorie les objets WebSphere MQ requis pour la réplication Q unidirectionnelle entre deux serveurs distants.

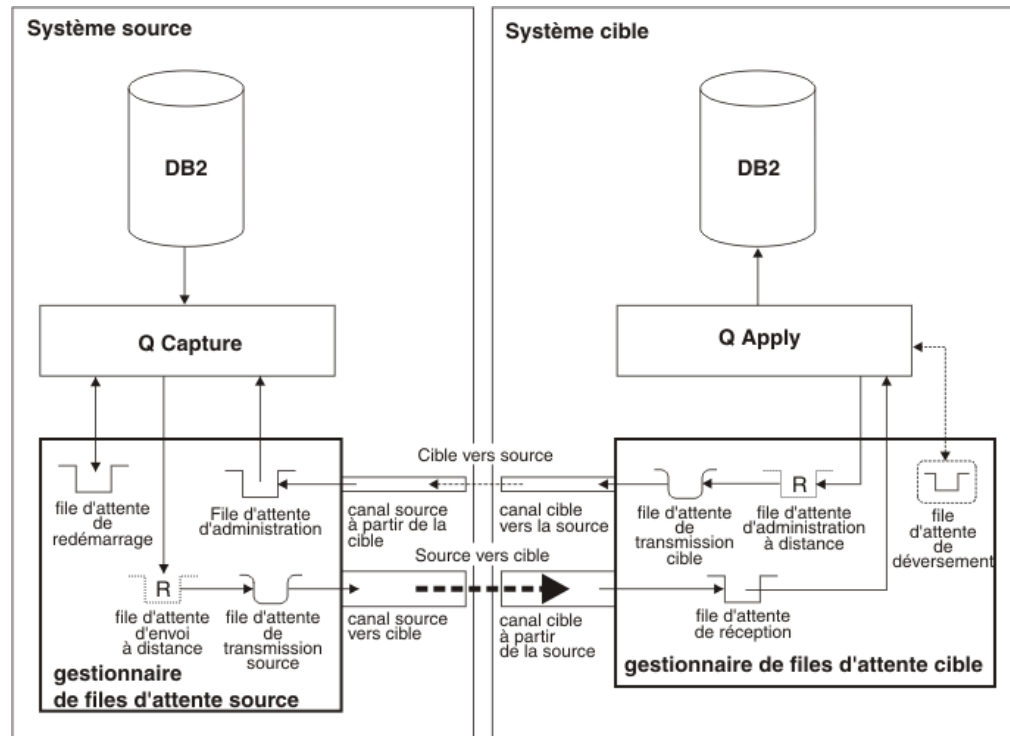


Figure 1. Objets WebSphere MQ requis pour la réplication Q unidirectionnelle entre deux serveurs distants. Les objets requis pour le programme Q Capture sont définis dans le gestionnaire de files d'attente du système source. Les objets requis pour le programme Q Apply sont définis dans le gestionnaire de files d'attente du système cible. Deux objets canal sont requis pour pouvoir créer une voie de transmission entre les systèmes source et cible, pour les messages de données et les messages d'informations issus du programme Q Capture. Deux objets sont également requis pour pouvoir créer une voie de transmission reliant le système cible au système source, pour les messages de données et les messages d'informations issus du programme Q Apply.

Objets WebSphere MQ nécessaires à la réplication unidirectionnelle (même système)

Lorsqu'un programme Q Capture effectue, sur le même système, une réplication de données vers un programme Q Apply, un seul gestionnaire de files d'attente est nécessaire. Vous pouvez utiliser la même file d'attente locale pour la file d'attente d'envoi et de réception ; les deux programmes peuvent partager une file d'attente d'administration locale.

Vous n'avez pas besoin de définitions de files d'attente distantes, de files d'attente de transmission ou de canaux.

La liste suivante illustre les objets WebSphere MQ requis pour la réplication Q unidirectionnelle ou pour la publication d'événements sur le même système :

- Un gestionnaire de files d'attente, utilisé par les deux programmes (Q Capture et Q Apply)
- Une file d'attente locale, utilisée à la fois comme file d'attente d'envoi et comme file d'attente de réception
- Une file d'attente locale, utilisée en tant que file d'attente d'administration pour les programmes Q Capture et Q Apply
- Une file d'attente locale, à utiliser comme file d'attente de redémarrage

- Une file d'attente modèle utilisée par Q Apply pour créer des file d'attente auxiliaires dynamiques afin de stocker les modifications apportées aux tables source pendant le processus de chargement de la table cible.

La figure 2 répertorie les objets WebSphere MQ requis pour la réplication Q unidirectionnelle sur le même système.

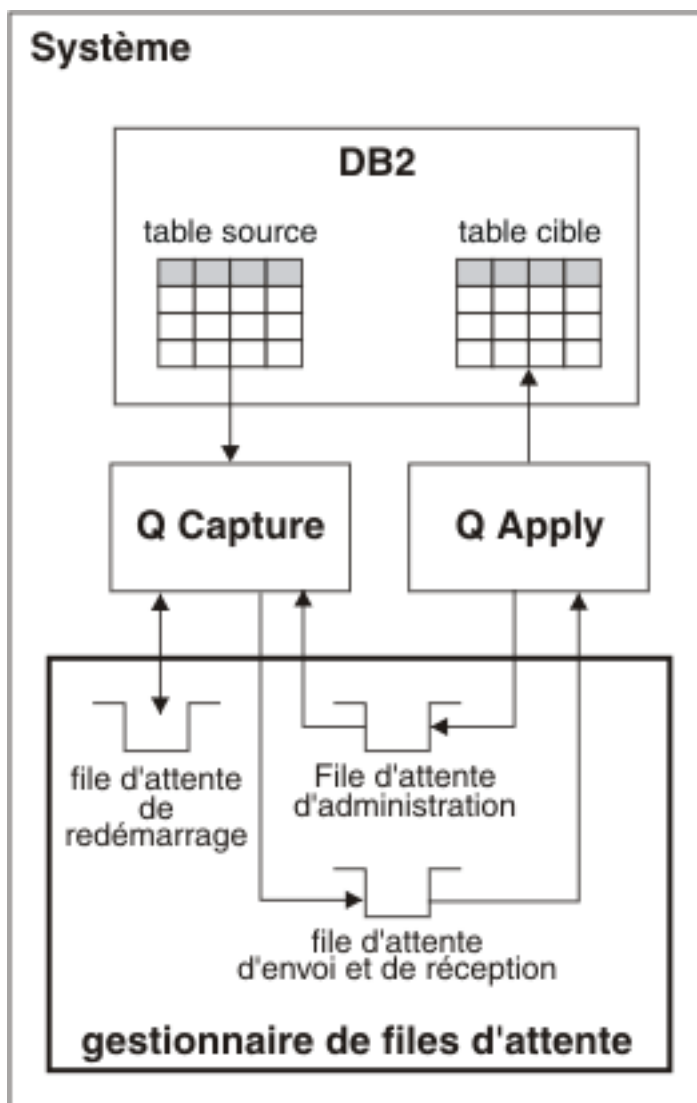


Figure 2. Objets WebSphere MQ nécessaires à la réplication Q unidirectionnelle sur le même système. Lorsque les programmes Q Capture et Q Apply sont exécutés sur le même système, un seul gestionnaire de files d'attente est requis. Une file d'attente locale est utilisée en tant que file d'attente d'envoi et de réception, et une autre file d'attente locale sert de file d'attente d'administration pour les programmes Q Capture et Q Apply.

Lors de la création de tables de contrôle pour le programme Q Capture et le programme Q Apply en réplication sur le même système, vous devez spécifier le même gestionnaire de files d'attente pour les deux jeux de tables de contrôle. Si vous créez une mappe de files de réplication, vous pouvez spécifier la même file d'attente locale pour les files d'attente d'envoi et de réception. La file d'attente d'administration spécifiée au moment de la création des tables de contrôle Q Capture peut également être spécifiée en tant que file d'attente d'administration Q Apply lorsque vous créez une mappe de files d'attente d'administration.

Objets WebSphere MQ requis pour la publication d'événements

La publication d'événements entre serveurs distants nécessite un gestionnaire de files d'attente et des files d'attente pour le programme Q Capture et pour l'application utilisateur. Les serveurs étant distribués, vous aurez également besoin de files d'attente de transmission et de canaux pour la transmission des données transactionnelles et pour la communication au sein d'un réseau.

Les listes suivantes répertorient les objets requis pour la publication d'événement entre deux serveurs distants :

Objets non canaux du système source

- Un gestionnaire de files d'attente
- Une file d'attente distante à utiliser comme file d'attente d'envoi (cette file d'attente pointe vers une file d'attente de réception sur le système cible)
- Une file d'attente locale à utiliser comme file d'attente d'administration
- Une file d'attente locale à utiliser comme file d'attente de redémarrage

Objets non canaux du système cible

- Un gestionnaire de files d'attente
- Une file d'attente locale à utiliser comme file d'attente de réception
- Une file d'attente distante à utiliser comme file d'attente d'administration (cette file d'attente pointe vers une file d'attente d'administration sur le système source)

Un canal de la source à la cible

- Canal émetteur défini au sein du gestionnaire de files d'attente source
- Une file d'attente de transmission locale associée
- Un canal récepteur de mise en correspondance défini au sein du gestionnaire de files d'attente cible

Un canal de la cible à la source

- Un canal émetteur défini au sein du gestionnaire de files d'attente cible
- Une file d'attente de transmission locale associée
- Un canal récepteur de mise en correspondance défini au sein du gestionnaire de files d'attente source

La figure 3, à la page 14 illustre les objets WebSphere MQ requis pour la publication d'événement entre des serveurs distants.

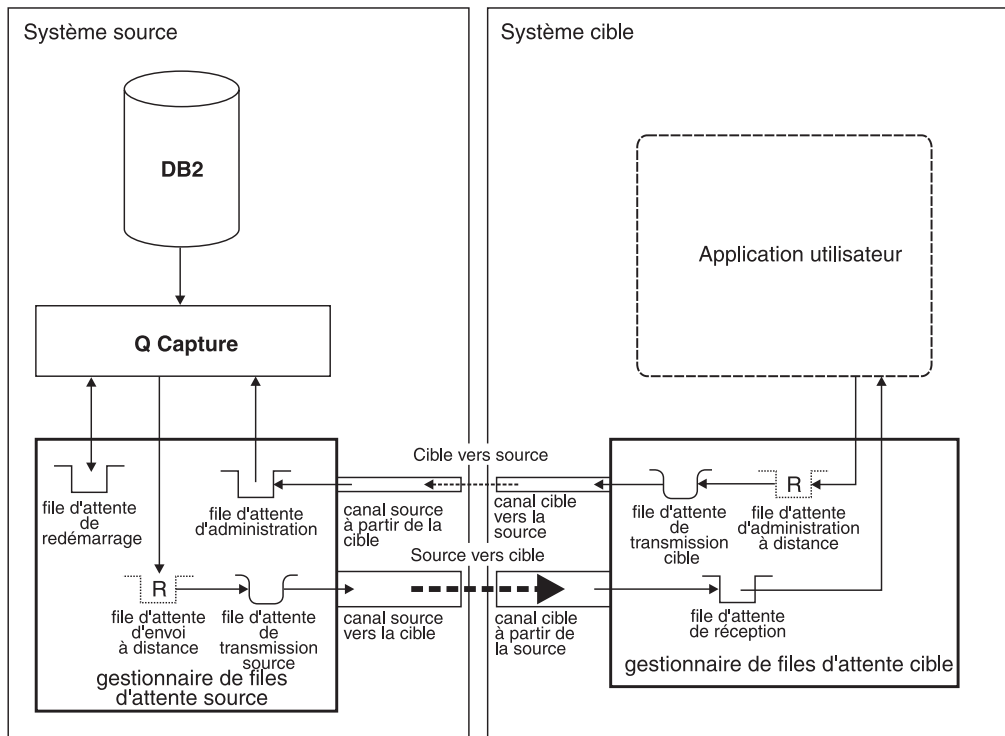


Figure 3. Objets WebSphere MQ requis pour la publication d'événement entre des serveurs distants. Les objets requis pour le programme Q Capture sont définis dans le gestionnaire de files d'attente du système source. Les objets requis pour l'application utilisateur sont définis dans le gestionnaire de files d'attente du système cible. Deux objets canal sont requis pour pouvoir créer une voie de transmission entre les systèmes source et cible, pour les messages de données et les messages d'informations issus du programme Q Capture. Deux objets canal sont également requis pour pouvoir créer une voie de transmission reliant le système cible au système source, pour les messages de contrôle issus de l'application utilisateur.

Si vous créez plusieurs canaux entre le programme Q Capture et l'application utilisateur, plusieurs files d'attente de transmission sont requises pour contenir les messages en attente de transfert.

Objets WebSphere MQ requis pour la réplication bidirectionnelle ou la réplication entre homologues (deux serveurs distants)

Pour répliquer des transactions, dans les deux sens, entre deux serveurs, vous devez définir deux jeux des mêmes objets WebSphere MQ requis pour la réplication unidirectionnelle. Une exception toutefois : un seul gestionnaire de files d'attente est obligatoire sur chaque système.

Par exemple, supposons que vous envisagiez de répliquer des transactions dans les deux sens entre le serveur A et le serveur B. Vous créez les objets WebSphere MQ qui relie le programme Q Capture du serveur A au programme Q Apply du serveur B. Vous devez également créer les objets WebSphere MQ qui relie le programme Q Capture du serveur B au programme Q Apply du serveur A. Les serveurs A et B se connectent à un seul gestionnaire de files d'attente sur les systèmes sur lesquels ils fonctionnent.

Les listes suivantes répertorient les objets requis pour la réplication bidirectionnelle ou entre homologues entre deux serveurs distants. Le gestionnaire de files d'attente

ne fait pas partie du serveur de réplication, mais est exécuté sur le même système ; par conséquent, les objets sont regroupés par système :

Objets non canal situés sur le système A

- Un gestionnaire de files d'attente
- Une définition de file d'attente éloignée à utiliser comme file d'attente d'envoi (cette file d'attente pointe vers une file d'attente de réception située sur le système B)
- Une file d'attente locale à utiliser comme file d'attente d'administration
- Une file d'attente locale à utiliser comme file d'attente de redémarrage
- Une file d'attente locale à utiliser comme file d'attente de réception
- Une définition de file d'attente éloignée à utiliser comme file d'attente d'administration (cette file d'attente pointe vers une file d'attente d'administration située sur le système B)
- Une définition de file d'attente modèle pour les files d'attente de déversement locales temporaires créée par le programme Q Apply et utilisée pendant le chargement des tables cible

Objets non canal situés sur le système B

- Un gestionnaire de files d'attente
- Une définition de file d'attente éloignée à utiliser comme file d'attente d'envoi (cette file d'attente pointe vers une file d'attente de réception sur le système A)
- Une file d'attente locale à utiliser comme file d'attente d'administration
- Une file d'attente locale à utiliser comme file d'attente de redémarrage
- Une file d'attente locale à utiliser comme file d'attente de réception
- Une définition de file d'attente éloignée à utiliser comme file d'attente d'administration (cette file d'attente pointe vers une file d'attente d'administration sur le système A)
- Une définition de file d'attente modèle pour les files d'attente de déversement locales temporaires créée par le programme Q Apply et utilisée pendant le chargement des tables cible

Objets canal

Objets canal transmis du système A vers le système B

- Un canal émetteur défini au sein du gestionnaire de files d'attente du système A
- Une file d'attente de transmission locale associée située sur le système A
- Un canal récepteur de mise en correspondance défini au sein du gestionnaire de files d'attente du système B

Objets canal transmis du système B vers le système A

- Un canal émetteur défini au sein du gestionnaire de files d'attente du système B
- Une file d'attente de transmission locale associée située sur le système B
- Un canal récepteur de mise en correspondance défini au sein du gestionnaire de files d'attente du système A

La figure 4, à la page 16 répertorie les objets WebSphere MQ requis pour la réplication Q bidirectionnelle ou entre homologues entre deux serveurs distants.

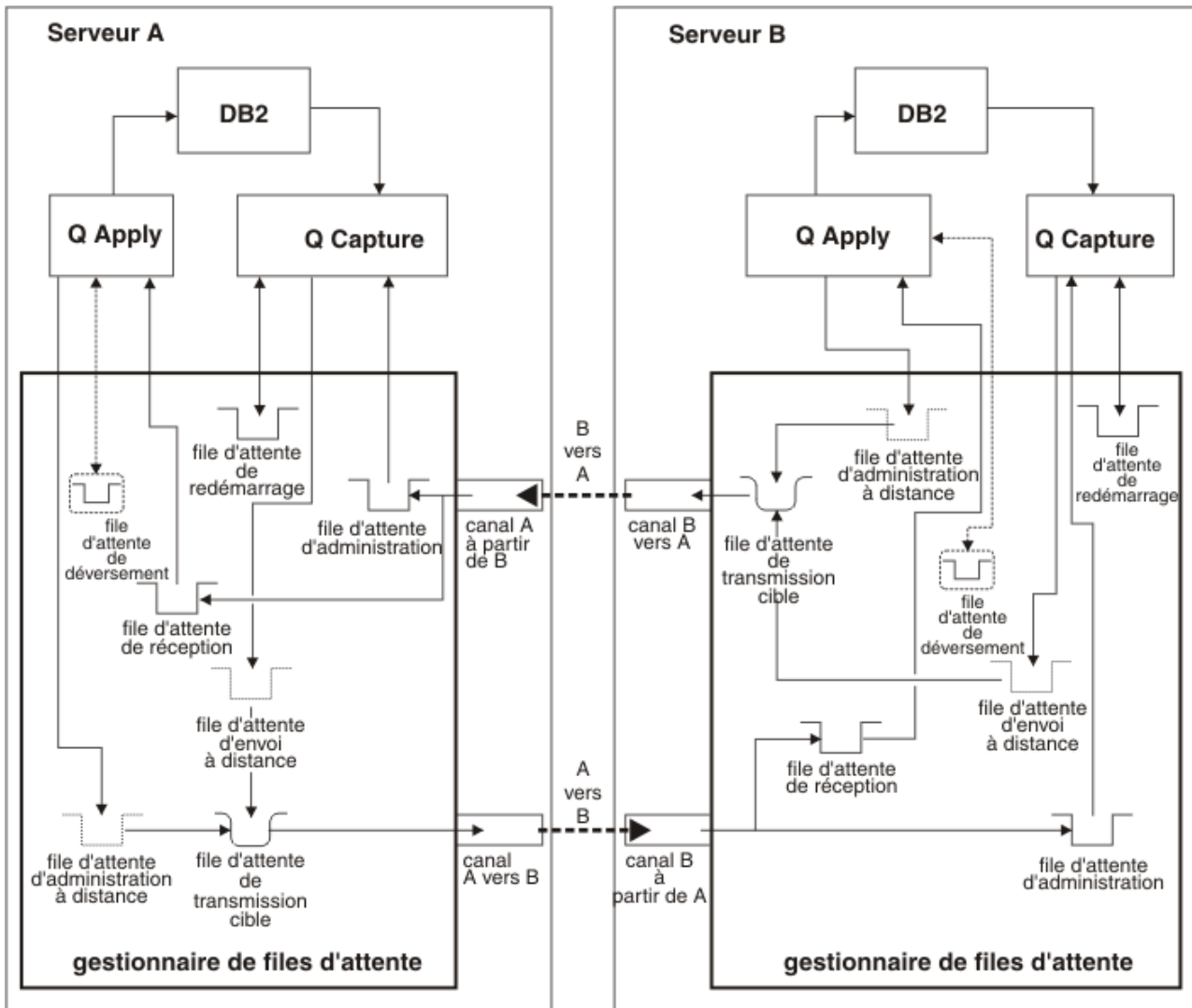


Figure 4. Objets WebSphere MQ requis pour la réplication Q bidirectionnelle ou la réplication entre homologues entre deux serveurs distants.. Vous devez créer deux jeux des mêmes objets WebSphere MQ requis pour la réplication Q unidirectionnelle entre un programme Q Capture et un programme Q Apply. Un jeu d'objets gère la réplication dans un sens tandis que l'autre jeu la gère en sens inverse. Un seul gestionnaire de files d'attente est requis sur chaque système.

Objets WebSphere MQ requis pour la réplication entre homologues (trois serveurs distants ou plus)

Dans un groupe d'homologues comptant trois serveurs distants ou plus, chaque serveur requiert un canal sortant vers chaque serveur supplémentaire du groupe. Chaque serveur requiert également un canal entrant partant de chaque serveur supplémentaire du groupe.

Au niveau de chaque serveur, le programme Q Apply nécessite une file d'attente d'administration distante par canal sortant. Il ne nécessite qu'une file d'attente d'administration locale, car tous les messages entrants provenant des programmes Q Apply sont gérés par un seul gestionnaire de files d'attente et acheminés vers une seule file d'attente.

Le nombre de files d'attente d'envoi et de réception dépend du nombre de serveurs dans le groupe.

Par exemple, dans un groupe à trois serveurs distants, le programme Q Capture du serveur A a besoin de deux files d'attente d'envoi, une pour les transactions à destination du serveur B et une pour celles à destination du serveur C. Le programme Q Apply du serveur A a besoin de deux files d'attente de réception, une pour les transactions en provenance du serveur B et une pour celles en provenance du serveur C.

Les listes suivantes répertorient les objets requis sur chaque système pour la réplication entre homologues avec trois serveurs ou plus :

Objets non canal situés sur chaque système

- Un gestionnaire de files d'attente
- Une file d'attente d'envoi distante pour chaque canal sortant
- Une file d'attente locale à utiliser en tant que file d'attente d'administration pour le programme Q Capture
- Une file d'attente locale, à utiliser comme file d'attente de redémarrage
- Une file d'attente de réception locale pour chaque canal entrant
- Une file d'attente d'administration distante pour le programme Q Apply pour chaque canal sortant
- Une définition de file d'attente modèle pour les files d'attente de déversement locales temporaires créée par le programme Q Apply et utilisée pendant le chargement des tables cible

Objets canal sortant situés sur chaque système

Créez ces objets pour chaque serveur supplémentaire du groupe. Par exemple, dans un groupe composé de trois serveurs, chaque serveur a besoin de deux canaux sortants.

- Un canal émetteur défini au sein du gestionnaire de files d'attente local
- Une file d'attente de transmission locale associée
- Un canal récepteur de mise en correspondance défini au sein du gestionnaire de files d'attente du serveur distant connecté à ce canal

Objets canal entrant situés sur chaque système

Créez ces objets pour chaque serveur supplémentaire du groupe. Par exemple, dans un groupe comptant trois serveurs, chaque serveur nécessite deux canaux entrants.

- Un canal récepteur défini au sein du gestionnaire de files d'attente local
- Un canal émetteur de mise en correspondance défini au sein du gestionnaire de files d'attente du serveur distant connecté à ce canal

La figure 5, à la page 18 illustre les objets WebSphere MQ requis sur un serveur inclus dans la réplication entre homologues (entre trois serveurs distants), lorsqu'une table logique est répliquée.

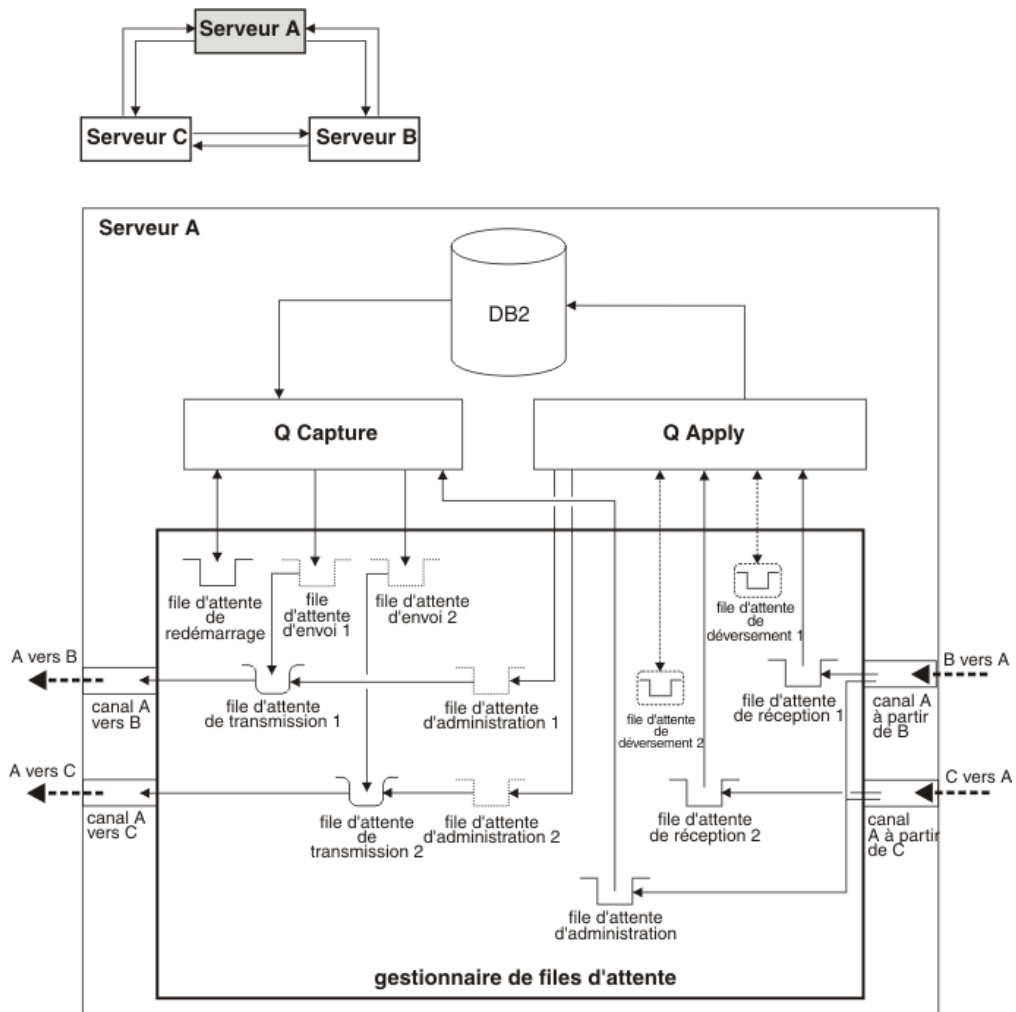


Figure 5. Objets WebSphere MQ requis sur un serveur inclus dans la réplication entre homologues avec deux autres serveurs distants. Sur chaque système, vous devez créer un gestionnaire de files d'attente. Le programme Q Capture requiert une file d'attente d'administration et une file d'attente de redémarrage. Vous devez créer une file d'attente d'envoi distante pour chaque canal sortant. Le programme Q Apply requiert une file d'attente d'administration distante par canal sortant, ainsi que d'une file d'attente de réception locale par canal entrant. Vous devez créer un canal entrant et un canal sortant par serveur supplémentaire au sein du groupe.

Paramètres requis pour les objets WebSphere MQ

Les objets WebSphere MQ utilisés par la réplication Q et la publication d'événements doivent posséder des propriétés spécifiques.

Cette rubrique décrit les paramètres requis pour les objets WebSphere MQ utilisés dans différents scénarios et se compose des sections suivantes :

- «Objets WebSphere MQ au niveau de la source», à la page 19
- «Objets WebSphere MQ au niveau de la cible», à la page 21

Recommandation : Créez les gestionnaires de file d'attente, les files d'attente et autres objets pour l'infrastructure de messagerie, avant de créer des objets de

réplication tels que les tables de contrôle et les abonnements Q, et avant de démarrer les programmes Q Capture et Q Apply. Les outils et exemples suivants sont disponibles pour vous aider :

- Les générateurs de script MQ que vous pouvez utiliser en cliquant sur l'icône du générateur de script dans la barre de menu du Centre de réplication
- La commande CREATE MQ SCRIPT du programme de ligne de commande ASNCLP
- **z/OS** L'exemple de travail ASNQDEFQ dans l'ensemble de données l'ensemble de données SASNSAMP
- **Linux UNIX Windows** L'exemple de script asnqdefq dans le répertoire SQLLIB/samples/rep1/q

Remarque sur les messages permanents : Depuis la version 9.7 (ou le PTF pour le niveau APAR PK85947 ou supérieur sous DB2 pour z/OS version 9.1), la réplication Q et la publication d'événements ne requièrent plus de messages WebSphere MQ permanents. Vous pouvez choisir d'utiliser des messages non permanents en définissant **message_persistence=n** lorsque vous démarrez le programme Q Capture ou en modifiant la valeur enregistrée de la colonne MESSAGE_PERSISTENCE dans la table IBMQREP_CAPPARMS. Les messages non permanents ne sont pas consignés et ne peuvent pas être récupérés. Le programme Q Capture envoie toujours des messages persistants à la file d'attente d'administration, quelle que soit la définition du paramètre **message_persistence**. Si vous créez des files d'attente non permanentes avec DEFPSIST(N), ces messages permanents remplacent le paramètre de la file d'attente.

Pour plus d'informations sur la configuration des objets WebSphere MQ, consultez le centre de documentation WebSphere MQ à l'adresse <http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp>.

Objets WebSphere MQ au niveau de la source

Le tableau 2 spécifie les valeurs requises applicables aux paramètres sélectionnés pour les objets WebSphere MQ au niveau de la source

Tableau 2. Valeurs de paramètres requises pour les objets WebSphere MQ au niveau de la source

Nom d'objet	Paramètres requis
Gestionnaire de files d'attente	<p>MAXMSGL</p> <p>(Taille de messages maximale autorisée dans les files d'attente de ce gestionnaire de files d'attente.) Cette valeur doit être au moins égale à la valeur de l'option max_message_size définie lors de la création d'une mappe de files de réplication ou de publication. Le paramètre max_message_size définit la mémoire tampon de messages allouée pour chaque file d'attente d'envoi. La valeur de l'option MAXMSGL doit également être au moins égale à celle de l'option MAXMSGL définie pour chaque file d'attente d'envoi, file d'attente de transmission et file d'attente d'administration.</p>
File d'attente d'envoi	<p>PUT (ENABLED)</p> <p>Permet au programme Q Capture de placer des messages de données et des messages d'informations dans la file d'attente.</p>

Tableau 2. Valeurs de paramètres requises pour les objets WebSphere MQ au niveau de la source (suite)

Nom d'objet	Paramètres requis
File d'attente d'administration	GET (ENABLED) Permet au programme Q Capture d'obtenir des messages de la file d'attente.
	PUT(ENABLED) Permet au programme Q Apply de placer des messages d'information dans la file d'attente.
	SHARE Plusieurs instances d'application peuvent obtenir des messages provenant de la file d'attente.
File d'attente de redémarrage	PUT (ENABLED) Permet au programme Q Capture de placer un message de redémarrage dans la file d'attente.
	GET (ENABLED) Permet au programme Q Capture d'obtenir les messages de redémarrage de la file d'attente.
File d'attente de transmission	USAGE(XMITQ) File d'attente de transmission.
	MAXDEPTH(500000) Le nombre maximal de messages autorisés dans la file d'attente de transmission est 999999. Il est peu probable que ce maximum soit nécessaire en raison de la vitesse à laquelle le programme Q Apply se maintient au niveau de Q Capture. Utilisez un paramètre qui reflète la charge de travail de transaction que vous prévoyez.
	MAXMSGL Vérifiez que la taille maximale des messages de la file d'attente n'est pas inférieure à la valeur de l'option MAXMSGL définie pour la file d'attente de réception du système cible, ni aux valeurs de max_message_size que vous avez définie lors de la création d'une mappe de files de réplication.
	Recommandation : Utilisez une file d'attente de transmission pour chaque paire de file d'attente envoi-réception.
Canal émetteur	CHLTYPE(SDR) Canal émetteur.
	DISCINT (0) Vérifiez que l'intervalle de déconnexion est suffisamment important pour éviter les dépassements de temps pendant les périodes ne comportant aucune transaction à répliquer.
	CONVERT(NO) Indique que l'agent MCA de transmission ne doit pas tenter de convertir les messages si l'agent MCA de réception ne peut pas exécuter cette conversion.
	HBINT Coordonnez cette valeur avec celle du paramètre heartbeat_interval de la mappe de files de réplication ou de publication. Si vous utilisez le paramètre HBINT pour envoyer des flux de pulsation, définissez heartbeat_interval sur 0 afin d'éliminer les messages de pulsation.

Tableau 2. Valeurs de paramètres requises pour les objets WebSphere MQ au niveau de la source (suite)

Nom d'objet	Paramètres requis
Canal récepteur	CHLTYPE(RCVR) Canal récepteur.

Objets WebSphere MQ au niveau de la cible

Le tableau 3 spécifie les valeurs requises applicables aux paramètres sélectionnés pour les objets WebSphere MQ au niveau de la cible.

Tableau 3. Valeurs de paramètres requises pour les objets WebSphere MQ au niveau de la cible

Nom d'objet	Paramètres requis
Gestionnaire de files d'attente	MAXMSGL (Taille de messages maximale autorisée dans les files d'attente de ce gestionnaire de files d'attente.) Cette valeur doit être au moins égale à la valeur de l'option max_message_size définie lors de la création d'une mappe de files de réplication. Le paramètre max_message_size définit la mémoire tampon de messages allouée pour chaque file d'attente d'envoi.
File d'attente de réception	GET(ENABLED) Permet au programme Q Apply d'obtenir des messages de la file d'attente. PUT(ENABLED) Permet au programme Q Capture de placer des messages de données et d'informations dans la file d'attente. MAXMSGL Vérifiez que la taille maximale des messages de la file d'attente est au moins égale à la valeur de l'option MAXMSGL définie pour la file d'attente de transmission du système source, et aux valeurs de max_message_size et de MEMORY_LIMIT définies lors de la création d'une mappe de files de réplication. MAXDEPTH(500000) Définissez le nombre maximal de messages autorisés dans la file d'attente de réception à un nombre qui reflète votre charge de travail de réplication. DEFSOPT(SHARED) Permet l'utilisation de plusieurs unités d'exécution Q Apply avec cette file d'attente. z/OS INDXTYPE(MSGID) Indique que les gestionnaires de files d'attente gèrent un index des messages en fonction de l'identificateur de message pour exécuter les opérations MQGET dans la file d'attente.
File d'attente d'administration	PUT(ENABLED) Permet au programme Q Apply ou à l'application utilisateur de placer des messages de contrôle dans la file d'attente.

Tableau 3. Valeurs de paramètres requises pour les objets WebSphere MQ au niveau de la cible (suite)

Nom d'objet	Paramètres requis
File d'attente de déversement modèle	<p>Nom de la file d'attente Par défaut, le programme Q Apply recherche la file d'attente modèle IBMQREP.SPILL.MODELQ. Vous pouvez spécifier un autre nom de file d'attente modèle à utiliser pour un abonnement Q lorsque vous créez ou modifiez l'abonnement Q.</p> <p>DEFTYPE(PERMDYN) Spécifie que les files d'attente de déversement sont des files d'attente permanentes dynamiques. Elles sont créées et supprimées sur demande du programme Q Apply, mais elles ne seront pas perdues si vous redémarrez le gestionnaire de files d'attente. Les messages sont consignés et peuvent être récupérés.</p> <p>DEFSOPT(SHARED) Permet à plusieurs unités d'exécution (différentes unités d'exécution agent et l'unité d'exécution agent de déversement) d'accéder simultanément aux messages de la file d'attente de déversement.</p> <p>MAXDEPTH(500000) Limite supérieure recommandée applicable au nombre de messages présents dans la file d'attente de déversement. Adaptez ce nombre en fonction du nombre de changements envisagés dans la table source pendant le chargement de la table cible.</p> <p>MSGDLVSQ(FIFO) Spécifie que les messages de la file d'attente de déversement sont fournis selon l'ordre premier entré, premier sorti.</p> <p>Remarque : Si vous avez plusieurs programmes Q Apply qui partagent le même gestionnaire de files d'attente, chaque programme Q Apply doit avoir sa propre file d'attente modèle avec un nom qui est unique sous ce gestionnaire de files d'attente.</p>
File d'attente de transmission	<p>USAGE(XMITQ) File d'attente de transmission.</p>
Canal émetteur	<p>CHLTYPE(SDR) Canal émetteur.</p> <p>DISCINT (0) Vérifiez que l'intervalle de déconnexion est suffisamment important pour éviter les dépassements de temps pendant les périodes d'inactivité lorsqu'un nombre réduit de messages doit être envoyé par le programme Q Apply ou par une application utilisateur.</p> <p>CONVERT(NO) Indique que l'agent MCA de transmission ne doit pas tenter de convertir les messages si l'agent MCA de réception ne peut pas exécuter cette conversion.</p>
Canal récepteur	<p>CHLTYPE(RCVR) Canal récepteur.</p>

Exemples de commandes de création d'objets WebSphere MQ pour la réplication Q et la publication d'événements

Vous pouvez utiliser les commandes du script WebSphere MQ (MQSC) et les commandes système pour créer les objets WebSphere MQ nécessaires à la réplication Q et à la publication d'événements.

Pour créer des gestionnaires de files d'attente, vous pouvez utiliser les commandes système suivantes :

Création de gestionnaires de files d'attente

```
crtmqm -lp 50 -ls 10 nom_gestionnaire_file
```

Cette commande crée un gestionnaire de file d'attente et spécifie qu'il utilise 50 fichiers journaux principaux et 10 fichiers journaux secondaires.

Lancement de gestionnaire de files d'attente

```
strmqm nom_gestionnaire_file
```

Une fois que vous avez créé et lancé un gestionnaire de files d'attente, vous pouvez utiliser les commandes MQSC dans le tableau 4 et le tableau 5, à la page 24 pour créer les objets.

Les tables contiennent les objets WebSphere MQ nécessaires pour définir la réplication unidirectionnelle. Vous pouvez utiliser les mêmes commandes avec quelques modifications mineures pour créer des objets pour la réplication multidirectionnelle ou la publication d'événements.

Utilisez la commande système `runmqsc nom_gestionnaire_file` pour commencer une session MQSC, puis exécutez les commandes MQSC dans la table de façon interactive, ou encore en créant des scripts exécutés sur chaque serveur.

Vous pouvez également utiliser les outils du générateur de script MQ du Centre de réplication pour créer des objets WebSphere MQ sur chaque serveur.

Objets utilisés par le programme Q Capture

Tableau 4. Exemples de commandes MQSC pour les objets WebSphere MQ utilisés par le programme Q Capture (suppose un gestionnaire de file d'attente source nommé CSQ1 et un gestionnaire de file d'attente cible nommé CSQ2)

Objet	Exemple de commande
File d'attente d'envoi File d'attente acheminant les messages de données d'un programme Q Capture. vers un programme Q Apply ou une application utilisateur. Dans les configurations distantes, cela correspond à la définition locale, sur le système source, de la file d'attente de réception située sur le système cible. Chaque file d'attente d'envoi doit être utilisée par un seul programme Q Capture.	<pre>DEFINE QREMOTE('ASN.SAMPLE_TO_TARGET.DATA') RNAME('ASN.SAMPLE_TO_TARGET.DATA') RQMNAME('CSQ2') XMITQ('CSQ2') PUT(ENABLED)</pre>

Tableau 4. Exemples de commandes MQSC pour les objets WebSphere MQ utilisés par le programme Q Capture (suppose un gestionnaire de file d'attente source nommé CSQ1 et un gestionnaire de file d'attente cible nommé CSQ2) (suite)

Objet	Exemple de commande
<p>File d'attente d'administration File d'attente locale qui reçoit des messages de contrôle à partir d'un programme Q Apply ou d'une application utilisateur destinés au programme Q Capture. Chaque file d'attente d'administration doit être lue par un seul programme Q Capture.</p>	<pre>DEFINE QLOCAL ('ASN.ADMINQ') PUT (ENABLED) GET (ENABLED) SHARE</pre>
<p>File d'attente de redémarrage File d'attente locale contenant un seul message, qui indique au programme Q Capture l'emplacement de début de lecture du fichier journal de reprise DB2 pour chaque file d'attente d'envoi après un redémarrage. Chaque programme Q Capture doit posséder sa propre file d'attente de redémarrage.</p>	<pre>DEFINE QLOCAL ('ASN.RESTARTQ') PUT (ENABLED) GET (ENABLED)</pre>
<p>File d'attente de transmission File d'attente locale qui met des messages en attente de transfert via un canal. Cette file d'attente peut être nommée pour le gestionnaire de files d'attente de destination (rappel de destination des messages). Recommandation : Utilisez une file d'attente de transmission pour chaque paire de file d'attente envoi-réception.</p>	<pre>DEFINE QLOCAL ('CSQ2') USAGE (XMITQ) MAXDEPTH (500000)</pre>
<p>Canal émetteur Extrémité émettrice du canal reliant le système source au système cible.</p>	<pre>DEFINE CHANNEL ('CSQ1.TO.CSQ2') CHLTYPE (SDR) CONNNAME ('IP_address (port)') TRPTYPE (TCP) XMITQ ('CSQ2') DISCINT (0) CONVERT (NO)</pre> <p>Où <i>adresse_IP</i> correspond à l'adresse IP du système cible et <i>port</i> représente un paramètre facultatif spécifiant un port inutilisé sur le système cible. Le port par défaut de WebSphere MQ est 1414.</p>
<p>Canal récepteur Extrémité réceptrice du canal reliant le système cible au système source.</p>	<pre>DEFINE CHANNEL ('CSQ2.TO.CSQ1') CHLTYPE (RCVR) TRPTYPE (TCP)</pre>

Objets utilisés par le programme Q Apply

Tableau 5. Exemples de commandes MQSC pour objets WebSphere MQ utilisés par le programme Q Apply

Objet	Exemple de commande
<p>File d'attente de réception File d'attente qui reçoit des messages de données et d'informations à partir d'un programme Q Capture destinés à un programme Q Apply ou une application utilisateur. Sur le système cible, cela correspond à une file d'attente locale.</p>	<pre>DEFINE QLOCAL ('ASN.SAMPLE_TO_TARGET.DATA') GET (ENABLED) PUT (ENABLED) DEFSOPT (SHARED) MAXDEPTH (500000)</pre>

Tableau 5. Exemples de commandes MQSC pour objets WebSphere MQ utilisés par le programme Q Apply (suite)

Objet	Exemple de commande
<p>File d'attente d'administration File d'attente qui achemine les messages de contrôle du programme Q Apply ou d'une application utilisateur vers un programme Q Capture . Dans les configurations distantes, cette file d'attente correspond à la définition locale, sur le système cible, de la file d'attente d'administration sur le système source.</p>	<pre>DEFINE QREMOTE('ASN.ADMINQ') RNAME('ASN.ADMINQ') RQMNAME('CSQ1') XMITQ('CSQ1') PUT(ENABLED)</pre>
<p>File d'attente auxiliaire File d'attente modèle que vous définissez sur le système cible pour conserver les messages des transactions envoyés par un programme Q Capture pendant le chargement d'une table cible. Le programme Q Apply crée des files d'attente de déversement de façon dynamique pendant le processus de chargement, conformément à la définition de file d'attente modèle, puis les supprime. Le nom par défaut de la file d'attente modèle est IBMQREP.SPILL.MODELQ. Lorsque vous créez ou modifiez un abonnement Q, vous pouvez spécifier l'utilisation d'une file d'attente modèle portant un nom différent, ainsi que l'utilisation d'une file d'attente modèle différente pour chaque abonnement Q.</p>	<pre>DEFINE QMODEL('IBMQREP.SPILL.MODELQ') DEFSOPT(SHARED) MAXDEPTH(500000) MSGDLVSQ(FIFO) DEFTYPE(PERMDYN)</pre>
<p>File d'attente de transmission File d'attente locale qui met des messages en attente de transfert via un canal. Cette file d'attente peut être nommée pour le gestionnaire de files d'attente de destination (rappel de destination des messages).</p>	<pre>DEFINE QLOCAL('CSQ1') USAGE(XMITQ)</pre>
<p>Canal émetteur Extrémité réceptrice du canal reliant le système cible au système source.</p>	<pre>DEFINE CHANNEL('CSQ2.TO.CSQ1') CHLTYPE(SDR) CONNNAME('IP_address (port)') TRPTYPE(TCP) XMITQ('CSQ1') DISCINT(0) CONVERT(NO)</pre> <p>Où <i>adresse_IP</i> correspond à l'adresse IP du système source et <i>port</i> représente un paramètre facultatif spécifiant un port inutilisé situé sur le système source. Le port par défaut de WebSphere MQ est 1414.</p>
<p>Canal récepteur Extrémité émettrice du canal reliant le système source au système cible.</p>	<pre>DEFINE CHANNEL('CSQ1.TO.CSQ2') CHLTYPE(RCVR) TRPTYPE(TCP)</pre>

Exécution des programmes de réplication sur un client WebSphere MQ

Vous pouvez exécuter les programmes Q Capture, Q Apply ou le moniteur d'alertes de réplication sur un système qui utilise un client WebSphere MQ pour vous connecter au gestionnaire de files d'attente utilisé par le programme de réplication.

Avant de commencer

- L'ID utilisateur de l'agent MCA (Message Channel Agent) WebSphere MQ doit correspondre à celui qui exécute les programmes de réplication sur le système client. L'ID utilisateur de l'agent MCA est défini à l'aide du paramètre MCAUSER dans la définition du canal de connexion serveur. Vous pouvez également spécifier MCAUSER(' '). Il s'agit de la valeur par défaut de z/OS SVRCONN2.
- **Linux UNIX Windows** Si vous avez changé le chemin d'accès par défaut de WebSphere MQ, vous devez modifier la variable d'environnement qui pointent vers les bibliothèques d'exécution de WebSphere MQ. Reportez-vous à la rubrique relative aux implications d'une gestionnaire de files d'attente 64 bits dans le centre de documentation de WebSphere MQ pour en savoir plus sur les meilleures pratiques pour la définition d'un chemin d'accès de bibliothèques.
- Vous devez définir un canal SVRCONN sur le système où s'exécute le gestionnaire de files d'attente, canal qui doit accepter les connexions client du système sur lequel s'exécutent le programme Q Capture ou Q Apply.
- Lorsque vous migrez un gestionnaire de files d'attente vers la version 7.1, des étapes supplémentaires pourraient s'avérer nécessaire pour activer une authentification de canaux. Pour plus d'informations, voir Channel authentication (authentification de canaux) dans le centre de documentation de WebSphere MQ.

Restrictions

- **z/OS** Les sous-systèmes WebSphere MQ pour z/OS ne peuvent pas être des clients.
- Vous ne pouvez pas utiliser le centre de réplication ou le programme ASNCLP pour répertorier les gestionnaires de file d'attente par défaut de Q Capture ou Q Apply lorsque ces programmes s'exécutent sur un client.

z/OS

A propos de cette tâche

Lorsque vous configurez un client WebSphere MQ, vous définissez les variables d'environnement qui pointent vers le système sur lequel le gestionnaire de files d'attente est exécuté ou vers une table de définition de canal client contenant des informations similaires. Vous définissez également une variable d'environnement afin de signaler aux programmes de réplication que le gestionnaire de files d'attente est sur un serveur. Lorsque ces variables sont définies, les programmes Q Capture, Q Apply ou le moniteur d'alertes de réplication chargent de manière dynamique les bibliothèques du client WebSphere MQ.

Pour plus d'informations sur la définition d'une configuration client-serveur WebSphere MQ, voir la rubrique relative à la procédure d'un client WebSphere MQ dans le centre de documentation WebSphere MQ.

Recommandation : Pour obtenir des performances optimales, exécutez les programmes Q Capture et Q Apply sur le même système que celui qui exécute le gestionnaire de files d'attente.

Procédure

Pour exécuter un programme de réplication Q sur un client WebSphere MQ :

1. Sur le système client, définissez la variable d'environnement de réplication `ASNUSEMQCLIENT=true`.
2. Définissez la connectivité client WebSphere MQ à l'aide de l'une des méthodes suivantes :

Méthode	Description
Variable d'environnement <i>MQSERVER</i>	<p>Définissez la variable d'environnement système <i>MQSERVER</i> pointant vers le gestionnaire de files d'attente utilisé par le programme de réplication. Vous pouvez définir cette variable de deux façons différentes selon que vous utilisez ou non une table de définition de canal client :</p> <p>Vous n'utilisez pas de table de définition de canal client : pointez vers le programme d'écoute du canal SVRCONN Vous pouvez indiquer le programme d'écoute sur le serveur WebSphere MQ qui contrôle les demandes du client dans le format suivant : <i>nom_canal/type_transport/hôte(port)</i>. Par exemple, pour faire pointer la variable <i>MQSERVER</i> vers le gestionnaire de files d'attente distant RQMGR2 avec un canal SVRCONN SYSTEM.DEF.SVRCONN sur l'hôte MQHOST avec le port 1414 :</p> <pre>MQSERVER="SYSTEM.DEF.SVRCONN/TCP/MQHOST(1414)"</pre> <p>Table de définition de canal client : nom du gestionnaire de files d'attente uniquement Si vous utilisez une table de définition de canal client, il vous suffit d'indiquer le nom du gestionnaire de files d'attente. Par exemple, si le programme de réplication fonctionne sur un système client WebSphere MQ et utilise le gestionnaire de files d'attente éloignées RQMGR2, définissez <i>MQSERVER=RQMGR2</i> sur le système client.</p>
Variables d'environnement <i>MQCHLLIB</i> ou <i>MQCHLTAB</i>	<p>Ces variables sont requises uniquement avec une table de définition du canal client. Définissez les variables d'environnement <i>MQCHLLIB</i> ou <i>MQCHLTAB</i> pour spécifier le chemin d'accès au fichier contenant la table de définition du canal client. Pour obtenir des informations détaillées, voir "MQCHLLIB" et "MQCHLTAB" dans le centre de documentation WebSphere MQ.</p> <p>Remarque : Dans la mesure où <i>MQCHLLIB</i> et <i>MQCHLTAB</i> ont des valeurs par défaut spécifiques à la plateforme, vous pouvez vous dispenser de définir ces variables d'environnement si le fichier de définition de canal client se trouve sur le système client dans le répertoire spécifié par la valeur par défaut de <i>MQCHLLIB</i> (par exemple, <code>/var/mqm/</code> sous Linux ou UNIX), et sous le nom de fichier que la valeur par défaut de <i>MQCHLTAB</i> spécifie (par exemple, <code>AMQCLCHL.TAB</code> sous Linux ou UNIX).</p>

Validation des objets WebSphere MQ pour la réplication Q et pour la publication d'événements

Vous pouvez utiliser le Centre de réplication pour afficher, sélectionner et valider les paramètres des gestionnaires de file d'attente WebSphere MQ et les files d'attente dans votre configuration. Le programme de ligne de commande `ASNCLP` peut également effectuer les contrôles de validation. Ces deux outils peuvent envoyer des messages test pour valider les objets dans une mappe de files de réplication.

Avant de commencer

- Le gestionnaire de files d'attente dans lequel les objets sont définis doit être lancé.
- **z/OS** Pour définir l'accès des administrateurs à WebSphere MQ, exécutez le modèle de travail ASNADMSP. Pour plus de détails, voir Activation des outils d'administration de la réplication pour utiliser WebSphere MQ.
- **Linux UNIX** Sous Linux et UNIX, ajoutez le chemin d'accès aux bibliothèques WebSphere MQ à la variable d'environnement DB2LIBPATH puis arrêtez et démarrez DB2 avant de démarrer le Centre de réplication. Le fait de ne pas définir de chemin d'accès peut provoquer un comportement inattendu, par exemple, le nombre de processus asnadmt peut augmenter alors qu'ils ne sont pas liés.
- **Linux UNIX Windows** La procédure stockée d'administration de la réplication installée pour prendre en charge cette fonction nécessite un espace table temporaire défini par l'utilisateur. Si aucun espace table temporaire défini par l'utilisateur n'a l'autorisation USE dans le groupe PUBLIC ou l'utilisateur en cours, le nouvel espace table ASNADMINIFSPACE est créé. L'utilisateur qui est connecté à la base de données doit être autorisé à créer la procédure stockée. Si un nouvel espace table est créé, l'utilisateur doit être autorisé à créer l'espace table temporaire et à autoriser l'utilisation de l'espace table pour ALL.
- Si les outils d'administration de la réplication fonctionnent sur un système client WebSphere MQ, vous devez rendre les variables d'environnement client-serveur disponibles pour DB2 de telle manière que la procédure stockée d'administration de l'application puisse y accéder. Pour plus d'informations, voir «Exécution des programmes de réplication sur un client WebSphere MQ», à la page 25.

A propos de cette tâche

Si vous utilisez la fonction de test de message, le Centre de réplication ou ASNCLP tente de mettre un message test dans la file d'attente d'envoi et obtient le message de la file d'attente de réception. Cet outil tente également de placer un message de test dans la file d'attente d'administration de Q Apply et d'obtenir le message envoyé par la file d'attente d'administration de Q Capture.

Restriction : Le test de message échoue si les programmes Q Capture ou Q Apply sont en cours d'exécution.

Procédure

1. Pour valider les objets WebSphere MQ de réplication Q et de publication d'événements, appliquez l'une des méthodes suivantes :

Méthode	Description
Centre de réplication	<ol style="list-style-type: none"> 1. Une fois que vous avez spécifié les objets WebSphere MQ dans une fenêtre ou dans l'assistant, cliquez sur Valider les objets WebSphere MQ pour vérifier que les propriétés des gestionnaires de files d'attente et des files d'attente sont correctes. Le Centre de réplication vérifie les objets et envoie des messages qui spécifient les paramètres à modifier. 2. Si vous créez ou modifiez une mappe de files de réplication, vous pouvez envoyer des messages test. Dans la fenêtre Validation des files d'attente WebSphere MQ, sélectionnez Envoyer des messages de test, puis cliquez sur Démarrer. <p>Des messages contenant les résultats des tests sont affichés dans la zone de messages de la fenêtre.</p>

Méthode	Description
Programme de ligne de commande ASNCLP	<p>1. Utilisez la commande VALIDATE WSMQ ENVIRONMENT FOR. Des messages contenant les résultats des tests sont envoyés à la sortie standard (stdout).</p> <p>Par exemple, pour valider les files d'attente d'envoi, de réception et d'administration Q Apply spécifiées pour la mappe de files de réplication SAMPLE_ASN_TO_TARGET_ASN :</p> <pre>VALIDATE WSMQ ENVIRONMENT FOR REPLQMAP SAMPLE_ASN_TO_TARGET_ASN</pre> <p>2. Si vous créez ou modifiez une mappe de files de réplication, utilisez la commande VALIDATE WSMQ MESSAGE FLOW FOR REPLQMAP. Des messages contenant les résultats des tests sont envoyés à la sortie standard (stdout).</p> <p>Par exemple, pour envoyer des messages de test entre les files d'attente spécifiées pour la mappe de files de réplication SAMPLE_ASN_TO_TARGET_ASN :</p> <pre>VALIDATE WSMQ MESSAGE FLOW FOR REPLQMAP SAMPLE_ASN_TO_TARGET_ASN</pre>

- Vous pouvez modifier le gestionnaire de file d'attente ou les files d'attente et utiliser de nouveau la fonction de validation pour vérifier vos modifications.

Pour plus d'informations sur les codes anomalie WebSphere MQ, consultez le centre de documentation WebSphere MQ à l'adresse <http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp>.

Contrôles de validation WebSphere MQ effectués par les outils de réplication

Lorsque vous utilisez le Centre de réplication ou le programme de ligne de commande ASNCLP pour valider votre configuration de WebSphere MQ, les outils peuvent contrôler si les files d'attente ont les propriétés adéquates et ils peuvent également envoyer des messages de test pour s'assurer que les données répliquées ou publiées peuvent être transmises.

Les outils utilisent une procédure stockée pour se connecter à WebSphere MQ et effectuer les tests de validation. Sous z/OS, vous devez installer la procédure stockée en utilisant l'exemple de travail ASNADMSP dans le jeu de données SASNSAMP. Sous Linux, UNIX, et Windows, la procédure stockée est automatiquement installée, mais des conditions requises existent pour son utilisation. Voir «Validation des objets WebSphere MQ pour la réplication Q et pour la publication d'événements», à la page 27 pour plus de détails.

Le tableau 6, à la page 30 décrit les contrôles de validation que les outils effectuent sur les objets WebSphere MQ eux-mêmes, et le tableau 7, à la page 30 décrit les contrôles de validation qui sont effectués pour s'assurer que les objets sont correctement définis dans les tables de contrôle.

Tableau 6. Contrôles de validation WebSphere MQ pour les propriétés de files d'attente valides

Message ASN	Code anomalie WebSphere MQ	Contrôle de validation
ASN2262E	2058	Le gestionnaire de file d'attente spécifié existe-t-il ?
ASN2263E	2059	Le gestionnaire de file d'attente est-il disponible pour la connexion ?
ASN2264E	2322 (Voir également Utilisation du serveur de commande.)	Le serveur de commande du gestionnaire de file d'attente est-il en cours d'exécution ?
ASN2267E	2085	La file d'attente existe-t-elle dans le gestionnaire de file d'attente spécifié ?
ASN2268E	Tous les codes anomalie non répertoriés.	WebSphere MQ a-t-il renvoyé l'un de ces codes anomalie ?
ASN2272W	2001, 2011, 2042, 2043, 2045, 2046, 2052, 2057, 2091, 2092, 2101, 2152, 2184, 2194, 2196, 2198, 2199, 2201	
ASN2273W	2082	La file d'attente d'alias spécifiée pointe-t-elle vers la file d'attente de base correcte ?
ASN2274W	Aucun. Voir Création d'une file d'attente de transmission.	La file d'attente de transmission existe-t-elle pour la file d'attente d'envoi distant ou la file d'attente d'administration de Q Apply ?
ASN2275W	Aucun. Voir Modification des attributs du gestionnaire de file d'attente ou Changement des attributs de file d'attente locale.	La taille de message maximale (MAXMSGL) de la file d'attente est-elle supérieure à la taille de message maximale (MAXMSGL) de son gestionnaire de file d'attente ?

Tableau 7. Contrôles de validation WebSphere MQ pour une configuration adéquate de la réplication

Message ASN	Contrôle de validation
ASN2276W	La taille de message maximale (MAXMSGL) de la file d'attente d'envoi est-elle inférieure ou égale à la taille de message maximale (MAXMSGL) de la file d'attente de réception ?
ASN2277W	Les files d'attente d'administration et de redémarrage de Q sont-elles des files d'attente locales ou des files d'attente d'alias qui font référence à des files d'attente locales ?

Tableau 7. Contrôles de validation WebSphere MQ pour une configuration adéquate de la réplication (suite)

Message ASN	Contrôle de validation
ASN2278W	La file d'attente d'envoi est-elle une file d'attente locale, une file d'attente distante, ou une file d'attente d'alias qui fait référence à une file d'attente locale ou distante ?
ASN2279W	La valeur de max_message_size de la mappe de file d'attente est-elle inférieure ou égale à la taille de message maximale (MAXMSGL) de la file d'attente d'envoi ?
ASN2281W	Si Q Capture et Q Apply partagent un gestionnaire de file d'attente, la file d'attente d'envoi et la file d'attente d'administration de Q Apply sont-elles définies comme files d'attente locales ou files d'attente d'alias qui font référence à des files d'attente locales ?
ASN2282W	Si Q Capture et Q Apply partagent un gestionnaire de file d'attente, la file d'attente d'envoi et de réception sont-elles les mêmes file d'attente locale ou files d'attentes d'alias qui font référence à la même file d'attente locale ? Les files d'attente d'administration sont-elles la même file d'attente local ou files d'attente d'alias qui font référence à la même file d'attente locale ?
ASN2284W	Si Q Capture et Q Apply utilisent des gestionnaires de file d'attente différents, la file d'attente d'envoi et la file d'attente d'administration de Q Apply sont-elles des files d'attente distantes ou des files d'attente d'alias qui font référence à une file d'attente locale ?
ASN2285W	Si vous avez spécifié une phase de chargement pour un abonnement Q, une file d'attente de modèle existe-t-elle pour la création de files d'attente auxiliaires ?
ASN2286W	La file d'attente de modèle est-elle correctement définie pour la réplication ?
ASN2287W	La file d'attente d'envoi, la file d'attente de réception et les files d'attente d'administration sont-elles correctement configurées pour la réplication ?

Exigences en matière de connectivité et d'autorisation pour les objets WebSphere MQ

Avant de pouvoir répliquer ou publier des données, vous devez configurer des connexions entre les gestionnaires de files d'attente des systèmes sur lesquels les programmes de réplication Q et de publication d'événements fonctionnent. Vous devez également vous assurer que les ID utilisateur qui permettent l'exécution des programmes de réplication et de publication sont autorisés à effectuer les actions requises au niveau des objets WebSphere MQ. Cette rubrique décrit les exigences de connectivité et d'autorisation.

Exigences de connectivité

Les gestionnaires de files d'attente de chaque système impliqué dans une réplication ou une publication doivent pouvoir s'interconnecter. Dans les environnements répartis, les programmes Q Capture, Q Apply et les applications utilisateur communiquent en se connectant aux gestionnaires de file d'attente et en envoyant des messages par des définitions de file d'attente, des files d'attente de transmission et des canaux distants.

La réplication Q et la publication d'événements prennent également en charge le regroupement qui permet à un groupe de gestionnaires de files d'attente de communiquer directement sur le réseau, sans avoir à utiliser des définitions de files d'attente distantes, des files d'attentes de transmission et des canaux.

Les connexions client-serveur pour lesquelles le gestionnaire de files d'attente est exécuté sur un système différent de celui du programme Q Capture ou Q Apply pour lequel il gère les files d'attente, sont également prises en charge.

Pour plus d'informations sur les différentes configurations de gestionnaires de files d'attente et sur la méthode de configuration des connexions pour chacun d'entre eux, voir *Intercommunication WebSphere MQ*.

Exigences d'autorisation

Les files d'attente WebSphere MQ constituent le principal moyen d'échange de données et de communication utilisé par les programmes Q Capture et Q Apply et ces programmes doivent pouvoir accéder aux données dans les files d'attente.

Lorsque vous créez des objets WebSphere MQ, vérifiez que les ID utilisateur permettant d'exécuter les programmes de réplication possèdent les autorisations appropriées pour effectuer les actions requises sur ces objets. La liste suivante résume ces exigences pour les programmes Q Capture, Q Apply et le moniteur d'alertes de réplication.

Remarque : Les administrateurs de la sécurité ajoutent les utilisateurs devant administrer WebSphere MQ sur le même groupe mqm. Ce qui inclut le superutilisateur sur les systèmes Linux et UNIX. Toute modification apportée aux droits ou à l'appartenance au groupe mqm group ne sera reconnue que lorsque le gestionnaire de files d'attente aura été redémarré, sauf si vous émettez la commande MQSC **REFRESH SECURITY** ou son équivalent en PCF (programmable command format).

Autorisations nécessaires pour le programme Q Capture

Les ID utilisateur qui permettent d'exécuter un programme Q Capture doivent posséder l'autorisation pour :

- Se connecter au gestionnaire de files d'attente (MQCONN ou MQCONNX) sur le système sur lequel le programme Q Capture est exécuté.
- Réaliser les actions suivantes au niveau de la file d'attente d'envoi : ouvrir (MQOPEN), rechercher des attributs (MQINQ), placer des messages (MQPUT), valider les messages (MQCMIT) et annuler les messages (MQBACK).
- Réaliser les actions suivantes au niveau de la file d'attente d'administration Q Capture : ouvrir (MQOPEN), rechercher des attributs (MQINQ) et extraire les messages (MQGET).

- Réaliser les actions suivantes au niveau de la file d'attente de redémarrage : ouvrir (MQOPEN), rechercher des attributs (MQINQ), placer les messages (MQPUT) et extraire les messages (MQGET).
- Réaliser les actions suivantes au niveau de la file d'attente de transmission : ouvrir (MQOPEN), placer les messages (MQPUT) et rechercher des attributs (MQINQ).

Autorisations nécessaires pour le programme Q Apply

Les ID utilisateur qui permettent d'exécuter un programme Q Apply doivent posséder l'autorisation pour :

- Se connecter au gestionnaire de files d'attente (MQCONN ou MQCONNX) sur le système sur lequel le programme Q Apply est exécuté.
- Réaliser les actions suivantes au niveau de la file d'attente de réception : ouvrir (MQOPEN), recherche d'attributs (MQINQ) et extraire les messages (MQGET).
- Réaliser les actions suivantes au niveau de la file d'attente d'administration Q Apply : ouvrir (MQOPEN), recherche d'attributs (MQINQ) et placer les messages (MQPUT).
- Exécutez les actions suivantes sur des files d'attente auxiliaires : ouvrir (MQOPEN), placer des messages (MQPUT), extraire des messages (MQGET), supprimer (dlt), changer (chg) et effacer (clr).

Autorisations requises pour le moniteur d'alertes de réplication

Si un moniteur d'alertes de réplication est utilisé pour surveiller le nombre de messages contenu dans la file d'attente de réception (alerte QAPPLY_QDEPTH) ou le nombre de messages contenu dans la file d'attente de déversement (alerte QAPPLY_SPILLQDEPTH), l'ID utilisateur permettant d'exécuter le moniteur doit avoir l'autorisation de se connecter au gestionnaire de files d'attente situé sur le système sur lequel le programme Q Apply fonctionne.

Pour les programmes Q Capture et Q Apply, l'ID utilisateur associé aux agents MCA doit avoir l'autorisation de :

- Se connecter au gestionnaire de files d'attente local.
- Réaliser les actions suivantes au niveau de la file d'attente de transmission locale : ouvrir (MQOPEN) et placer les messages (MQPUT).

Pour plus d'informations sur les autorisations et les privilèges WebSphere MQ, voir *Sécurité WebSphere MQ*.

Exigences de stockage pour WebSphere MQ pour la réplication Q et la publication d'événements

Planification des ressources WebSphere MQ pour atteindre le niveau souhaité de résistance aux indisponibilités réseau ou aux indisponibilités de la cible, ou encore des deux. Si les messages ne peuvent pas être transportés, une quantité plus élevée de ressources est utilisée au niveau de la source. Si les messages ne peuvent pas être appliqués, une quantité plus élevée de ressources est utilisée au niveau de la cible.

Par défaut, les messages utilisés dans la réplication Q et la publication d'événements sont permanents. WebSphere MQ enregistre tous les messages permanents dans des fichiers journaux. Si vous redémarrez un gestionnaire de file d'attente après un incident, ce dernier extrait tous les messages consignés dont il a

besoin. Les messages permanents nécessitent davantage d'espace de stockage. Vous pouvez également choisir d'utiliser des messages non permanents en définissant `message_persistence=n` lorsque vous démarrez le programme Q Capture.

Pour plus d'informations sur les fichiers journaux WebSphere MQ, consultez le centre de documentation WebSphere MQ à l'adresse <http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp>.

Taille des messages WebSphere MQ

Vous pouvez limiter la taille des messages WebSphere MQ lorsque vous créez des files d'attente et des gestionnaires de files d'attente, ainsi que lors de la configuration de la réplication ou de la publication. Vous devez coordonner les limites de taille de message entre WebSphere MQ, la réplication Q et la publication d'événements.

Dans WebSphere MQ, vous définissez l'option MAXMSGL (longueur maximale des messages) pour limiter la taille des messages.

La liste suivante décrit la façon dont MAXMSGL fait référence aux limites de mémoire applicables aux programmes Q Capture et Q Apply.

Programme Q Capture

Vous pouvez limiter la quantité de mémoire utilisée par un programme Q Capture pour mettre un message en mémoire tampon avant de l'envoyer vers une file d'attente d'envoi. Vous définissez l'option MAX_MESSAGE_SIZE lorsque vous créez une mappe de files de réplication ou de publication. La valeur par défaut est 64 Ko.

Important : si vous spécifiez pour le programme Q Capture une mémoire tampon de taille plus élevée que celle que les files d'attentes peuvent gérer d'après la configuration, la réplication ou la publication ne peuvent pas fonctionner. Si la file d'attente d'envoi est éloignée de la file d'attente de réception, la taille MAX_MESSAGE_SIZE définie pour la mappe de files d'attente de réplication ou la mappe de files d'attente de publication stockée dans la table IBMQREP_SENDQUEUES doit être inférieure de 4 Ko à la valeur d'attribut MAXMSGL de la file d'attente de transmission et du gestionnaire de files d'attente. Ces 4 Ko représentent les informations supplémentaires figurant dans l'en-tête de message lorsque le message se trouve dans la file d'attente de transmission. Si les files d'attente d'envoi et de réception sont définies dans le même gestionnaire de files d'attente, vous pouvez utiliser la même valeur pour MAX_MESSAGE_SIZE et MAXMSGL les files d'attente.

Programme Q Apply

Vous pouvez limiter la quantité de mémoire utilisée par un programme Q Apply pour mettre en mémoire tampon plusieurs messages en provenance d'une file d'attente de réception avant que les unités d'exécution agent ne reconstituent les messages en transactions. Vous définissez l'option MEMORY_LIMIT lorsque vous créez une mappe de files de réplication. La valeur par défaut est 2 Mo.

Avertissement : Assurez-vous que la valeur de l'option MAXMSGL applicable à la file d'attente locale à utiliser comme file d'attente de réception n'est pas plus élevée que celle de l'option MEMORY_LIMIT applicable à la mappe de files de réplication contenant la file d'attente de réception.

Segmentation des messages

Le programme Q Capture divise automatiquement en plusieurs messages les transactions dont la taille dépasse la valeur de `MAX_MESSAGE_SIZE`, en décomposant la transaction au niveau des fins de ligne. Si vous répliquez ou publiez des données à partir de colonnes d'objets LOB d'une table source, le programme Q Capture divise automatiquement les données LOB en plusieurs messages. Cela permet de vérifier que les messages ne dépassent pas la valeur de l'option `MAX_MESSAGE_SIZE` définie pour la mappe de files de réplication ou de publication contenant chaque file d'attente d'envoi.

Sur certains systèmes d'exploitation, WebSphere MQ permet de définir une segmentation de messages, de telle sorte que les messages trop volumineux pour les files d'attente ou les canaux sont automatiquement divisés. La réplication Q et la publication d'événements n'utilisent pas cette fonction de segmentation de messages. Si vous configurez les objets WebSphere MQ en vue d'une utilisation de la segmentation de messages, vous devez quand même vérifier que la valeur de l'option `MAXMSGL` pour les files d'attente est supérieure ou égale à celle de l'option `MAX_MESSAGE_SIZE` pour la mappe de files de réplication ou de publication.

Pour plus d'informations sur la taille des messages, voir le Guide d'administration système *WebSphere MQ* correspondant à votre plateforme.

Considérations relatives à la longueur de file d'attente pour les valeurs d'objets LOB

Les valeurs d'objets LOB d'une table source sont susceptibles de dépasser la quantité maximale de mémoire allouée par un programme Q Capture pour chaque file d'attente d'envoi, sous forme de mémoire tampon de messages.

La valeur par défaut de `MAX_MESSAGE_SIZE` (mémoire tampon de messages) d'une file d'attente d'envoi est égale à 64 kilo-octets. Dans DB2, les valeurs d'objets LOB peuvent atteindre 2 Go ; par conséquent, les valeurs d'objets LOB sont souvent divisées en un grand nombre de messages.

Si vous envisagez d'effectuer une réplication de données d'objets LOB, vérifiez que la valeur affectée à `MAXDEPTH` pour la file d'attente de transmission et la file d'attente d'administration sur le système source, et pour la file d'attente de réception sur le système cible, sont suffisantes pour ce nombre de messages. Vous pouvez diminuer le nombre de messages requis pour l'envoi de données d'objets LOB en augmentant la valeur affectée à `MAX_MESSAGE_SIZE` pour la file d'attente d'envoi, au moment de la création ou de la publication d'une mappe de files de réplication.

Si une valeur élevée d'objet LOB est divisée dans une mémoire de messages relativement réduite, cela crée un très grand nombre de messages d'objet LOB susceptible de dépasser la quantité maximale de messages (`MAXDEPTH`) définie pour une file d'attente de transmission ou de réception. Cette situation entraîne l'affichage d'une erreur de file d'attente. Lorsqu'une file d'attente de réception distante comporte une erreur, l'agent MCA du système cible envoie un rapport d'exception WebSphere MQ pour chaque message qu'il ne peut pas envoyer. Ces rapports d'exception peuvent être conservés dans la file d'attente d'administration du programme Q Capture.

Regroupement de gestionnaires de files d'attente dans la réplication Q et la publication d'événements

La réplication Q fonctionne dans un environnement de gestionnaires de files d'attente regroupés. Le regroupement permet à un groupe de gestionnaires de files d'attente de communiquer directement sur le réseau sans devoir utiliser des définitions de files d'attente distantes, de files d'attentes de transmission et de canaux.

Une configuration en clusters permet de créer plusieurs instances d'une file d'attente portant le même nom sur différents gestionnaires de files d'attente au sein du même groupe. Toutefois, dans la réplication Q, la file d'attente de réception du système cible ne doit être définie qu'une seule fois au sein d'un groupe. Les programmes Q Capture et Q Apply utilisent un système dense de numérotation pour identifier et extraire les messages manquants. (Chaque message est associé à un nombre entier positif sans écart le suivant.) Les noms de files d'attente de réception doivent être uniques pour une paire donnée de programmes Q Capture et Q Apply. Si deux files d'attente de réception possèdent le même nom, le système de numérotation ne fonctionne pas.

La réplication Q ne fonctionne pas en association avec les listes de distribution des regroupements, qui utilisent simplement la commande MQPUT pour envoyer un même message à plusieurs destinations.

Pour plus d'informations, voir *Regroupement des gestionnaires de files d'attente WebSphere MQ*.

Chapitre 4. Configuration des bases de données pour la réplication Q et la publication d'événements (Linux, UNIX et Windows)

Avant de répliquer des données, vous devez définir des variables d'environnement et configurer les bases de données source et cible. Pour la publication d'événements, vous devez configurer uniquement la base de données source.

Obligatoire : Définition de DATA CAPTURE CHANGES sur les tables source DB2 et les tables système DB2 for z/OS

Vous devez définir l'attribut DATA CAPTURE CHANGES sur n'importe quelle table que vous souhaitez répliquer. De plus, avec DB2 for z/OS Version 9 et ultérieur, vous devez définir DATA CAPTURE CHANGES sur les tables de catalogue système SYSIBM.SYSTABLES, SYSIBM.SYSCOLUMNS et SYSIBM.SYSTABLEPART.

A propos de cette tâche

La définition de DATA CAPTURE CHANGES sur des tables source ordonne à DB2 de consigner les modifications SQL dans un format étendu requis pour la réplication. Les outils d'administration de réplication vont générer le langage de définition de données pour modifier la table si cette option n'est pas définie. Cependant, vous pouvez le définir lors de la création de tables ou de la modification de la table elle-même.

Pour les tables système DB2 for z/OS, la définition de DATA CAPTURE CHANGES active la détection et la réplication des modifications dans la structure des tables source, comme l'ajout de nouvelles colonnes ou les modifications des types de données de colonne.

Remarque : Si la source de réplication est DB2 for z/OS Version 9 ou ultérieure, le programme Q Capture s'arrête si DATA CAPTURE CHANGES n'est pas défini sur les tables de catalogue système SYSIBM.SYSTABLES, SYSIBM.SYSCOLUMNS et SYSIBM.SYSTABLEPART.

L'activation de DATA CAPTURE CHANGES pour une table induit un petit volume de consignation supplémentaire. Lorsque cette option est définie pour une table, DB2 consigne les images complètes avant et après de la ligne pour chaque mise à jour. Lorsque l'option n'est pas définie, DB2 consigne uniquement les colonnes qui ont changé car ce volume de consignation est le seul élément nécessaire pour DB2. Le volume de consignation supplémentaire est proportionnel à la taille de ligne mais uniquement pour les tables dont l'attribut DATA CAPTURE CHANGES est activé.

Restrictions

z/OS Si la table a été modifiée, vous ne pouvez pas définir DATA CAPTURE CHANGES sur cette table ou toute autre table du même espace table tant que ce dernier n'a pas été réorganisé.

Procédure

L'instruction CREATE TABLE ou ALTER TABLE permet de définir DATA CAPTURE CHANGES sur les tables source de réplication. Utilisez l'instruction ALTER TABLE pour définir DATA CAPTURE CHANGES sur les tables du catalogue système DB2 for z/OS.

Lorsque vous créez des tables de contrôle avec les outils d'administration de réplication de la version 10 sous z/OS, les outils vérifient l'attribut DATA CAPTURE CHANGES sur les tables système requises. Si l'attribut n'est pas défini, le code SQL généré pour la création de tables de contrôle comprend les instructions ALTER TABLE suivantes :

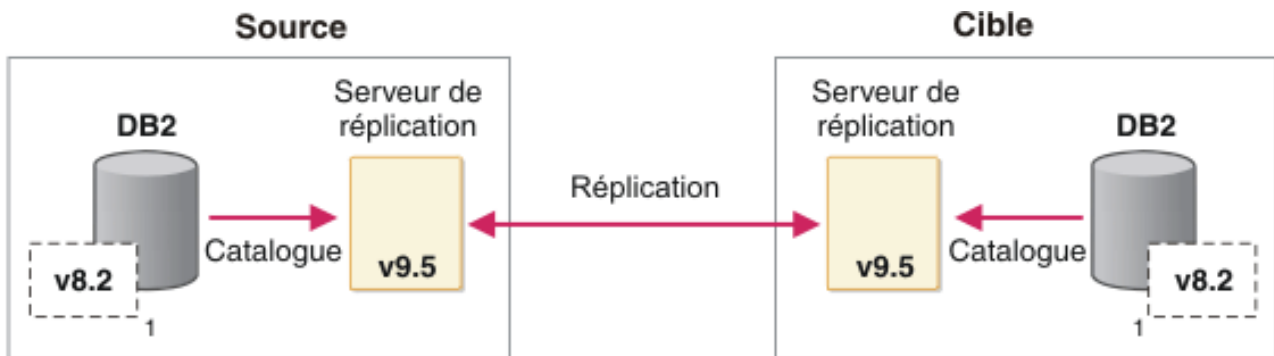
```
ALTER TABLE SYSIBM.SYSTABLES DATA CAPTURE CHANGES
ALTER TABLE SYSIBM.SYSCOLUMNS DATA CAPTURE CHANGES
ALTER TABLE SYSIBM.SYSTABLEPART DATA CAPTURE CHANGES
```

Si vous ne créez pas de tables de contrôle, vous pouvez utiliser ces mêmes instructions pour définir l'attribut requis.

Configuration pour les bases de données DB2 distantes ou antérieures

Vous pouvez configurer InfoSphere Replication Server ou Data Event Publisher avec des bases de données DB2 antérieures afin de pouvoir utiliser les fonctionnalités de réplication les plus récentes sans devoir migrer ces bases de données existantes. Vous pouvez également configurer InfoSphere Replication Server ou Data Event Publisher sur un autre serveur que ceux hébergeant vos bases de données DB2. Les deux configurations utilisent des étapes similaires.

La figure suivante illustre la configuration de base pour utiliser InfoSphere Replication Server avec une version antérieure de DB2. Dans cet exemple, il s'agit de InfoSphere Replication Server version 9.5 avec DB2 version 8.2.



(1) Cette configuration est actuellement valide avec les versions DB2, à partir de la version 8.2

Figure 6. Configuration de Replication Server avec des bases de données DB2 antérieures

La figure suivante illustre la configuration de base pour utiliser des bases de données DB2 distantes.

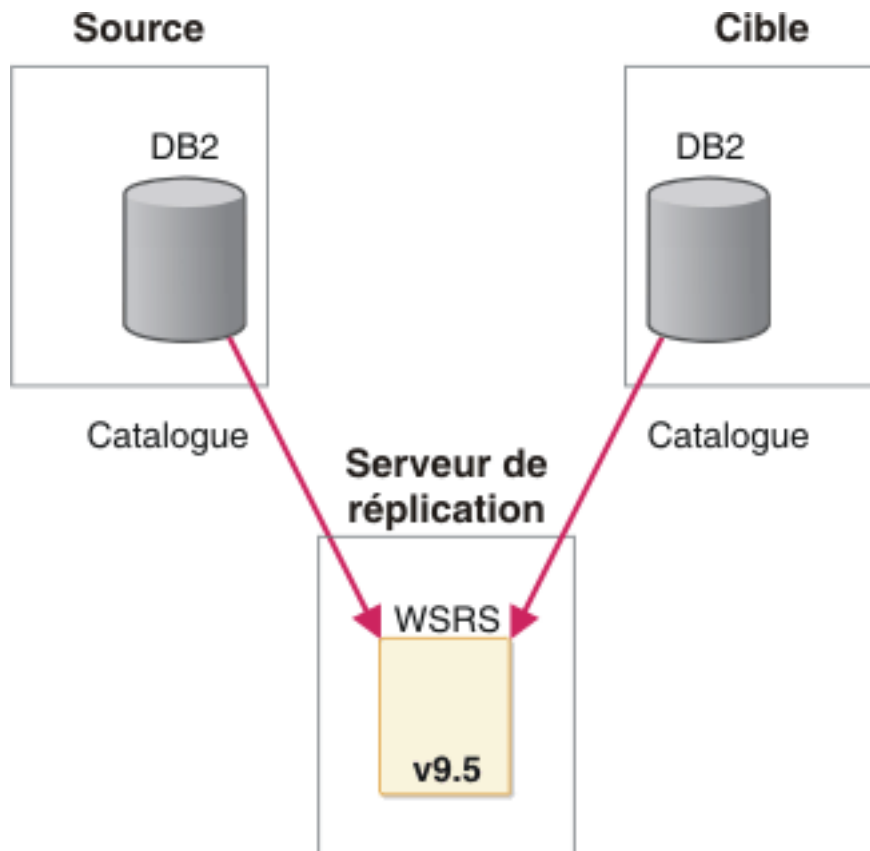


Figure 7. Configuration avec des bases de données DB2 distantes

Avant de commencer

Si vous prévoyez d'utiliser une nouvelle version de Replication Server ou de Data Event Publisher sur le même système qu'une base de données DB2 antérieure, voir Scénarios d'installation pour connaître les informations importantes à prendre en compte pour l'installation.

Si vous prévoyez d'utiliser un programme Q Capture avec une version ou une édition différente de celle votre programme Q Apply, voir Prise en charge de la coexistence de la réplication Q et de la publication d'événements version 9.7

A propos de cette tâche

L'installation de Replication Server ou de Data Event Publisher ne met pas à niveau vos bases de données DB2 existantes, même si la version de DB2 est antérieure. Après installation et acceptation de la licence de InfoSphere Replication Server ou de Data Event Publisher, les programmes de réplication pourront interagir avec toute base de données DB2 de niveau d'édition égale ou inférieure (jusqu'à la version 8.2), que cette base de données réside sur une autre instance DB2 ou sur un autre système. Dans le cadre de cette installation, une nouvelle instance DB2 est créée, mais cette instance DB2 n'a aucun besoin d'être démarrée et il est inutile d'y créer une base de données.

Les configurations à versions mixtes ou réplication à distance nécessitent des étapes supplémentaires :

- Les bases de données ou noeuds DB2 antérieurs ou distants doivent être catalogués sur l'instance contenant les programmes de réplication.
- Dans les configurations à versions mixtes, les tables de contrôle de réplication doivent être créées pour correspondre au niveau de version plus élevé des programmes de réplication.
- Dans les configurations à versions mixtes, le propriétaire de l'instance DB2 de plus haut niveau installée avec les tout derniers programmes de réplication doit disposer des droits en lecture sur les journaux DB2 de la version antérieure de DB2.

Restrictions

Les outils d'administration de réplication ne créent pas de tables de contrôle pré-version 10.1 pour une base de données source ou cible sous DB2 for Linux, UNIX et Windows version 10.1.

Pour Q Capture ou Q Apply qui sont locaux à DB2

- La réplication depuis des tables compressées ou partitionnées sur DB2 pour Linux, UNIX et Windows est uniquement prise en charge pour DB2 version 9.7 ou ultérieure. DB2 décompresse les enregistrements des journaux avant de les transmettre au programme Q Capture. Les données décompressées sont transmises de la source vers la cible sur laquelle la compression n'est pas nécessaire.
- Pour répliquer des opérations de chargement, la base de données source et le programme Q Capture doivent avoir le correctif APAR PK78558 ou un niveau de correctif plus récent sous z/OS ou ils doivent posséder la version 9.7 ou ultérieure sous Linux, UNIX et Windows.

Pour Q Capture ou Q Apply qui sont distants de DB2

- Le programme Q Capture, le programme Q Apply et les bases de données source et cible doivent être sur la même plateforme Linux, UNIX ou Windows et être dotés de la même architecture binaire, par exemple 32/32 ou 64/64 bits.
- Vous ne pouvez pas utiliser les outils d'administration de réplication pour répertorier ou valider des files d'attente.
- La valeur endian et la page de codes du système sur lequel sont exécutés les programmes Q Capture doivent correspondre à la valeur endian et à la page de codes de la base de données source et au système d'exploitation.

Procédure

1. Installez Replication Server ou Data Event Publisher et créez une instance comme vous y invite le programme d'installation. Vous pouvez également installer les produits sur le même système qu'une version antérieure de DB2 (jusqu'à la version j8.2) ou bien les installer sur un système distant.
2. Cataloguez le noeud et la base de données DB2 source ou cible sur l'instance nouvellement installée à l'aide des commandes **CATALOG NODE** et **CATALOG DATABASE**. Ouvrez une fenêtre de commande DB2 à partir de l'instance DB2 utilisée par la réplication ou par la publication d'événements et cataloguez le noeud et la base de données de la base de données distante ou antérieure.
3. Créez un fichier de mot de passe asnpwd à utiliser par les programmes de réplication et de publication d'événements.

4. Si la base de données source ou cible est une version antérieure à celle utilisée par Q Capture ou Q Apply, utilisez ASNCLP ou le Centre de réplication pour créer des tables de contrôle qui correspondent à la version des programmes de réplication.
5. Utilisez le Centre de réplication ou ASNCLP pour créer les mappes de files d'attente et les abonnements Q.
6. Si la base de données doit être une source de réplication, accordez les droits d'accès SYSADM ou DBADM à l'utilisateur que le programme Q Capture utilisera pour la connexion à la source. Le programme Q Capture pourra ainsi extraire les enregistrements du journal DB2.
7. Démarrez Q Capture et Q Apply à partir de l'instance de réplication, qu'il s'agisse d'une version nouvelle ou de la même version sur un système distant. Par exemple, démarrez la réplication à partir de l'instance nouvellement installée en fournissant l'alias de la base de données DB2 antérieure ou distante spécifiée dans la commande **CATALOG DATABASE**.

Configuration pour GDPS Active/Active (z/OS)

La réplication Q permet de synchroniser les données pour les sites GDPS Active/Active et propose une interface de gestion réseau (NMI) pour la surveillance de l'environnement de réplication à partir de IBM® Tivoli NetView pour z/OS.

Les sections suivantes décrivent la configuration et la sécurité pour la réplication Q avec les sites GDPS Active/Active.

Nom de la charge de travail

Pour les sites Active/Active, le nom de la charge de travail correspond au nom de la mappe de file d'attente de réplication identifiant les files d'attente d'envoi et de réception des messages entre sites. Une mappe de file d'attente de réplication est requise entre les programmes Q Capture et Q Apply appariés sur chaque site. Les deux charges de travail qui constituent les deux directions de la configuration bidirectionnelle doivent posséder le même nom de charge de travail et, par conséquent, le même nom de mappe de file d'attente. Ainsi, NetView est en mesure d'identifier la paire de charges de travail.

Le nom de mappe de file d'attente de réplication ne doit pas dépasser 64 caractères en raison de la longueur maximale autorisée pour le nom de la charge de travail. Le nom de mappe de file d'attente doit commencer par un caractère alphabétique mais les autres caractères peuvent être des lettres ou des chiffres. Le seul caractère spécial autorisé est le soulignement (_).

La réplication Q rapporte des mesures de charge de travail pour les charges de travail qui sont actives au moment de la demande de mesure par le biais de l'interface NMI (Network Management Interface). Aucune mesure n'est rapportée pour les charges de travail qui sont inactives ou qui étaient arrêtées depuis la dernière demande. Pour les mesures rapportées en tant que compteurs incrémentiels dans les espaces de travail NetView, la réplication Q rapporte à la fois les compteurs incrémentiels basés sur l'intervalle d'échantillonnage entre demandes NMI successives et les totaux cumulés pour le compteur depuis le démarrage de la charge de travail.

Utilisation de NMI pour la surveillance de la réplication

Vous pouvez utiliser Tivoli NetView for z/OS pour surveiller l'état de l'environnement de réplication Q entre des sites dans une solution de sites GDPS Active/Active. Tivoli Monitoring fournit une vue approfondie et unifiée des performances et de l'état des différents produits de réplication qui font partie de la solution GDPS A/A (réplication IMS, réplication Q, assistant de l'équilibreur de charge (Load Balancer Advisor)). Pour obtenir plus d'informations détaillées sur les performances, le réglage et le traitement des incidents des réplications, vous devez généralement utiliser le tableau de bord de réplication Q.

Une surveillance à partir de Tivoli est réalisée via une nouvelle interface, Network Management Interface, qui est un protocole de demande/réponse qui utilise un socket AF_UNIX. GDPS A/A ne requiert pas les tables de surveillance de Q Capture ou de Q Apply ; en revanche, il utilise cette nouvelle interface pour émettre des demandes NMI à des programmes Q Capture et Q Apply pour obtenir des mesures de performances.

L'interface NMI est implémentée par les programmes Q Apply et Q Capture. Elle permet aux clients NMI de demander des mesures d'exploitation comprenant les informations d'état et de performances, telles que le délai de réplication. Pour activer cette interface, vous devez définir les paramètres **nmi_enable** et **nmi_socket_name** pour Q Capture et Q Apply.

nmi_enable

Le paramètre **nmi_enable** indique si le programme Q Capture ou Q Apply est activé pour fournir une interface NMI (Network Management Interface) destinée à la surveillance des statistiques de la réplication Q à partir de Tivoli NetView. L'application client NMI doit être sur le même système z/OS que le programme Q Capture ou Q Apply.

nmi_socket_name

Le paramètre **nmi_socket_name** indique le nom du socket AF_UNIX où le programme Q Capture ou le programme Q Apply écoute les demandes d'informations statistiques provenant des applications client NMI. Vous pouvez indiquer ce paramètre pour modifier le nom de socket généré automatiquement par le programme.

Pour plus d'informations sur ces deux paramètres, consultez les rubriques suivantes :

- «Descriptions des paramètres asnqcap», à la page 375
- «Descriptions des paramètres asnqapp», à la page 417

Lorsque vous démarrez Q Capture ou Q Apply avec le paramètre **nmi_enable** ou le paramètre **nmi_socket_name**, les valeurs sont utilisées durant la session en cours de ces programmes. Pour sauvegarder les valeurs de ces paramètres d'une session à l'autre, vous devez mettre à jour les tables de paramètres respectives, à savoir IBMQREP_CAPPARMS pour Q Capture et IBMQREP_APPLYPARMS pour Q Apply. Chacune de ces tables comprend une colonne NMI_ENABLE et une colonne NMI_SOCKET_NAME, décrites dans les rubriques suivantes :

- «Table IBMQREP_CAPPARMS», à la page 494
- «Table IBMQREP_APPLYPARMS», à la page 538

SQL permet de mettre à jour la valeur de ces paramètres dans les tables de contrôle. Par exemple :

```
UPDATE schéma.IBMQREP_CAPPARMS SET NMI_ENABLE=Y
```

Où *schéma* est le nom du schéma des tables de contrôle pour cette instance Q Capture.

Sécurité

Lors de la connexion d'un client NetView, le programme Q Capture ou Q Apply authentifie le client auprès de la fonction de sécurité installée sur le système, par exemple RACF, TOP SECRET ou ACF2. Les programmes utilisent le protocole suivant :

- Le programme obtient l'identificateur de sécurité pour le client connecté grâce à un appel `ioctl()` de la commande SECIGET.
- L'appel de RACROUTE vérifie l'autorisation du client.
- Un client est considéré comme étant autorisé lorsque l'identificateur de sécurité possède un accès READ sur les profils suivants dans la classe de ressources SERVERAUTH :
 - QREP.NETMGMT.sysname.procname.QAPPLY.DISPLAY
 - QREP.NETMGMT.sysname.procname.QCAPTURE.DISPLAY
- Si le client ne peut pas accéder à ce profil, la réplication Q effectue un contrôle d'autorisation pour vérifier si le client possède les privilèges de superutilisateur (IUD égal à zéro ou autorisation d'accès à la ressource BPX.SUPERUSER dans la classe FACILITY). Dans ce cas, le client est autorisé.

Définition des variables d'environnement (Linux, UNIX, Windows)

Vous devez définir des variables d'environnement avant d'exécuter le programme Q Capture, le programme Q Apply ou le moniteur d'alertes de réplication, avant d'utiliser le programme de ligne de commande ASNCLP ou le Centre de réplication ou avant d'émettre des commandes système.

Procédure

Pour définir les variables d'environnement :

1. Définissez la variable d'environnement pour le nom d'instance DB2 (DB2INSTANCE) qui contient le serveur Q Capture, le serveur Q Apply et le serveur de contrôle de moniteur.
 - **Linux UNIX** Pour Linux et UNIX, utilisez la commande suivante :

```
export DB2INSTANCE=nom_instance_db2
```
 - **Windows** Pour Windows, utilisez la commande suivante :

```
SET DB2INSTANCE=nom_instance_db2
```
2. Si vous avez créé la base de données source avec une page de codes autre que la valeur de la page de codes par défaut, définissez la variable d'environnement DB2CODEPAGE sur cette page de codes.
3. Facultatif : Définissez la variable d'environnement pour la base de données DB2 par défaut (DB2DBDFT) sur le serveur Q Capture ou le serveur Q Apply.
4. **Linux UNIX Windows** Assurez-vous que les variables système comprennent le répertoire dans lequel les fichiers exécutables et les bibliothèques du programme de réplication Q sont installés. Dans le répertoire de base de l'instance DB2, le chemin d'accès de la bibliothèque par défaut est `SQLLIB/lib` et le chemin d'accès de l'exécutable par défaut est `SQLLIB/bin`. Si vous avez déplacé les bibliothèques ou les fichiers exécutables, mettez à jour vos variables d'environnement pour inclure le nouveau chemin d'accès.

5. **AIX et DB2 Extended Server Edition** : définissez la variable d'environnement EXTSHM sur ON dans les bases de données source et cible sous AIX ou uniquement dans la base de données source sous DB2 Extended Server Edition (si le programme Q Capture doit se connecter à plusieurs partitions de base de données), en entrant les commandes suivantes :

```
$ export EXTSHM=ON
$ db2set DB2ENVLIST=EXTSHM
```

- a. Assurez-vous que la variable d'environnement EXTSHM est définie à chaque démarrage de DB2. Pour ce faire, modifiez le fichier `/home/instancedb2/sqllib/profile.env` où `instancedb2` est le nom de l'instance DB2 contenant la base de données cible. Dans le fichier, ajoutez ou modifiez la ligne : `DB2ENVLIST='EXTSHM'`
- b. Ajoutez la ligne suivante au fichier `/home/instancedb2/sqllib/userprofile` : `export EXTSHM=ON`

Définition de la variable d'environnement TMPDIR Linux, UNIX)

Vous pouvez spécifier un répertoire pour que les programmes Q Capture et Q Apply écrivent des fichiers temporaires. L'écriture de ces fichiers dans un répertoire spécifié peut les protéger d'une suppression accidentelle.

A propos de cette tâche

Par défaut, les programmes de réplication utilisent le répertoire `/tmp` pour les fichiers temporaires. Dans certains cas, ces fichiers peuvent être supprimés par d'autres programmes ayant un droit d'accès à la racine. Par exemple, les administrateurs système Linux ou UNIX exécutent généralement des travaux de service basés sur l'heure pour supprimer des fichiers dans le répertoire `/tmp`.

Les fichiers temporaires manquants peuvent empêcher les programmes de communiquer. Par exemple, si vous émettez la commande `asnlqcmd stop` pour arrêter le programme Q Apply et qu'un fichier temporaire est manquant, la commande échoue.

Pour éviter toute suppression accidentelle, vous pouvez utiliser la variable d'environnement `TMPDIR` pour spécifier un répertoire temporaire.

Remarque : Les ID utilisateur qui exécutent les programmes de réplication et de publication doivent avoir un accès en écriture soit au répertoire `/tmp`, soit au répertoire spécifié par la variable `TMPDIR`.

Procédure

Pour définir la variable d'environnement `TMPDIR`, indiquez un répertoire accessible à l'ID utilisateur qui exécute les programmes de réplication ou de publication. Veillez à ce que les fichiers ne puissent pas être supprimés par d'autres ID utilisateur.

Par exemple, la commande suivante indique le répertoire `/home/repldba/tempfiles/` :

```
export TMPDIR=/home/repldba/tempfiles/
```

Résolution des problèmes de mémoire pour Q Capture et Q Apply (AIX)

Les systèmes d'exploitation AIX, vous permettent de garantir que les programmes de réplication disposent d'assez de mémoire pour fonctionner en définissant des variables d'environnement en plus des paramètres appropriés pour la réplication Q, WebSphere MQ et DB2.

A propos de cette tâche

Définir les variables d'environnement *EXTSHM* et *LDR_CNTRL* et utiliser la commande **ulimit** peut permettre d'éviter des problèmes de mémoire. Vous pouvez également augmenter la valeur du paramètre MAXAGENTS pour DB2.

Procédure

Pour résoudre les problèmes de mémoire pour Q Capture et Q Apply :

1. Vérifiez que les variables d'environnement AIX suivantes ont la valeur maximale :

EXTSHM

La variable *EXTSHM* (mémoire partagée étendue) supprime la limite de 11 zones de mémoire partagée uniquement. Les processus 64 bits ne sont pas affectés par la variable *EXTSHM*.

- a. Définissez *EXTSHM* sur ON dans les bases de données source et cible ou dans la base de données source uniquement si le programme Q Capture doit se connecter à des partitions de base de données multiples, en saisissant la commande suivante :

```
$ export EXTSHM=ON
$ db2set DB2ENVLIST=EXTSHM
```

- b. Vérifiez que la variable *EXTSHM* est définie à chaque démarrage de DB2. Pour ce faire, modifiez le fichier `/home/instancedb2/sql11b/profile.env`, où *instancedb2* est le nom de l'instance DB2 contenant la base de données cible. Dans le fichier, ajoutez ou modifiez la ligne : `DB2ENVLIST='EXTSHM'`
- c. Ajoutez la ligne suivante au fichier `/home/db2inst/sql11b/userprofile` :

```
export EXTSHM=ON
```

LDR_CNTRL

Vous pouvez utiliser la variable d'environnement *LDR_CNTRL* pour configurer le programme Q Apply pour l'utilisation de pages volumineuses pour ses données et ses segments de mémoire.

Remarque : Prenez des précautions avant de définir cette valeur. Si besoin est, vérifiez auprès du support AIX ou de votre administrateur système avant d'effectuer cette modification du système AIX.

Par exemple, la première commande ajoute quatre segments et la deuxième commande trois segments :

```
export LDR_CNTRL=MAXDATA=0x40000000
```

```
export LDR_CNTRL=MAXDATA=0x30000000
```

Commande **ulimit**

La commande **ulimit** définit des limites de ressources de processus

utilisateur ou génère sur celles-ci, tel que défini dans le fichier `/etc/security/limits`. La valeur `ulimit -a` pour l'instance DB2 indique que les données et les segments ne sont pas définis sur la valeur illimitée. Pour les instances 64 bits sur des serveurs AIX, il n'y a pas de contrainte sur le mémoire partagée des bases de données.

Recommandation : Vérifiez que tous les paramètres **ulimit** sont définis avec une valeur illimitée. Les données/segments sont parfois limités et ne peuvent pas être définis avec une valeur illimitée. Dans ce cas, consultez l'administrateur système.

2. Vérifiez que les paramètres WebSphere MQ sont corrects. Si WebSphere MQ est un client 32 bits (Version 5.3) et que AIX et DB2 sont 64 bits, la base de données DB2 contenant les tables de contrôle Q Capture ou Q Apply doit être cataloguée à l'aide d'une connexion de bouclage. Si vous utilisez WebSphere MQ V6.0 et DB2 32 bits, vous devez également définir une connexion de bouclage. Le bouclage est coûteux (le protocole TCP/IP est coûteux) mais si rien ne fonctionne, essayez la connexion de bouclage pour résoudre des incidents tels que SQL1224N.
3. Vérifiez que les paramètres DB2 sont corrects.

- a. Vérifiez **db2set** et assurez-vous que la variable `EXTSHM` est exportée.
- b. Recherchez d'éventuelles erreurs provoquées par les limites appliquées aux processus, aux unités d'exécution ou à l'espace de permutation du système d'exploitation. Ce type d'erreur peut être identifié par un message SQL1225N. Si vous recevez ce message, vérifiez la valeur actuelle du paramètre **maxuproc** qui définit le nombre maximal de processus par ID utilisateur. La commande suivante vérifie la valeur du paramètre **maxuproc** :

```
lsattr -E -l sys0
```

Définissez **maxuproc** sur une valeur supérieure en exécutant la commande suivante en tant qu'utilisateur root :

```
chdev -l sys0 -a maxuproc='nouvelle_valeur'
```

- c. Augmentez la valeur du paramètre MAXAGENTS dans la configuration du gestionnaire de base de données.

Lorsque le nombre d'agents atteint la valeur du paramètre MAXAGENTS, toutes les demandes suivantes nécessitant un nouvel agent sont refusées jusqu'à ce que le nombre d'agents passe au-dessous de la valeur. Cette valeur s'applique au nombre total d'agent, y compris les agents de coordination, les sous-agents, les agents inactifs et les agents en veille, fonctionnant sur toutes les applications. La valeur du paramètre MAXAGENTS doit être au minimum égale à la somme de la valeur du paramètre MAXAPPLS de chaque base de données pouvant être accédées simultanément.

4. Vérifiez que le paramètre **memory_limit** d'une mappe de files d'attente de réplication est défini sur 32 Mo (valeur par défaut) ou une valeur inférieure. Il s'agit de la valeur actuellement recommandée pour chaque file d'attente de réception.

Configuration de la base de données source en vue d'une interaction avec le programme Q Capture (Linux, UNIX, Windows)

Si la consignation avec conservation des journaux n'est pas activée au niveau de la base de données source, vous devez l'activer pour empêcher l'écrasement des entrées de journaux avant qu'un programme Q Capture ne les lise. Pour que cette modification soit prise en compte, vous devez également effectuer une sauvegarde hors connexion de la base de données source.

A propos de cette tâche

Important : la sauvegarde d'une base de données volumineuse peut prendre beaucoup de temps. Pendant ce processus, les applications sont déconnectées de la base de données et de nouvelles connexions ne sont pas autorisées.

Procédure

Vous pouvez configurer la base de données source à partir du Centre de réplication ou de la ligne de commande :

Pour configurer une base de données DB2 afin d'exécuter le programme Q Capture, utilisez l'une des méthodes suivantes :

Méthode	Description
Centre de réplication	<p>Utilise la fenêtre Activation de la consignation avec conservation des journaux pour activer la consignation avec conservation des journaux. Pour ouvrir cette fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture à activer et sélectionnez Activation de la base de données pour la réplication.</p> <p>Avertissement : si vous cochez la case Utilisation de l'assistant de configuration de la consignation, vous devez sélectionner le bouton d'option Consignation avec conservation des journaux sur la page Type de consignation. La seule option de l'assistant qui affecte la réplication est l'option d'activation de la consignation avec conservation des journaux.</p>

Méthode	Description
Ligne de commande	<ol style="list-style-type: none"> Vérifiez la valeur de "Log retain for recovery status" dans la configuration de la base de données. Si elle est définie sur NO, activez la consignation d'archivage en réglant la valeur du paramètre de configuration de base de données LOGARCHMETH1 sur une valeur autre que OFF. Facultatif : utilisez la commande update database configuration pour accroître les valeurs de la base de données source selon vos besoins pour la réplication ou la publication. Les valeurs suivantes de configuration de bases de données sont appropriées pour de nombreux grands scénarios de réplication (si la base de données est configurée en vue d'une interaction avec d'autres applications, vos valeurs peuvent alors être plus grandes que celles-ci) : APPLHEAPSZ 4096, LOGFILSIZ 2000, LOGPRIMARY 20, LOCKLIST 200, DBHEAP 1000, STMTHEAP 4096, LOGBUFSZ 64, MAXAPPLS 300. Il peut être nécessaire d'ajuster la valeur de LOCKTIMEOUT si vous rencontrez des verrouillages pendant que Q Capture duplique des données volumineuses LOB ou XML. Pour le paramètre LOGSECOND, la valeur 20 est adaptée à la plupart des scénarios. Si vous comptez répliquer de grandes transactions, il est recommandé de définir LOGSECOND = -1 sur DB2 version 8.1 ou ultérieure pour éviter la plupart des problèmes d'espace de consignation. Emettez la commande backup database à l'aide des paramètres appropriés pour la base de données source.

Configuration de la base de données cible en vue d'une interaction avec le programme Q Apply (Linux, UNIX et Windows)

Le programme Q Apply est un processus hautement parallèle que vous pouvez configurer afin de répondre à une variété de charges de travail de réplication. Selon la manière dont vous configurez le programme Q Apply, vous devez vous assurer que le paramètre MAXAPPLS (nombre maximum d'applications actives) est défini correctement. Le programme Q Apply utilise plusieurs agents en fonction du nombre que vous définissez pour diviser la charge de travail des transactions applicables vers les cibles. La base de données traite chaque agent comme une application distincte essayant de se connecter.

Avant de commencer

UNIX Pour les environnements 64 bits : si vous exécutez la réplication Q ou la publication d'événements dans un environnement 64 bits sur des plateformes HP-UX ou Solaris, cataloguez la base de données (le serveur Q Apply) comme base de données en boucle et créez une entrée pour cette base de données dans le fichier des mots de passe. Si vous ne cataloguez pas le serveur Q Apply en tant que base de données en boucle, un incident Semaphore Wait, sqlcode 1224, se produira lorsque le programme Q Apply atteindra la limite de la mémoire partagée.

Procédure

Pour définir le paramètre MAXAPPLS basé sur un scénario de réplication donné :

- Lancez la commande suivante : `update database configuration for database using MAXAPPLS n` où database est la base de données cible et n est le nombre

maximum d'applications autorisées pour exécuter simultanément la base de données. Pour déterminer la valeur de n, utilisez la formule suivante :

$$n \geq (\text{nombre d'applications autres que le programme Q Apply pouvant utiliser la base de données en même temps}) + (3 * \text{le nombre de programmes Q Apply sur la base de données}) + (\text{nombre de files d'attente de réception, chacune avec une unité d'exécution du navigateur} + \text{le nombre total d'agents Q Apply pour toutes les files d'attente de réception})$$

Voici un exemple des résultats de ce calcul basé sur un scénario où deux applications autres que le programme Q Apply peuvent utiliser la base de données en même temps. Il y a deux programmes Q Apply, un qui utilise 3 agents pour traiter une seule file d'attente de réception et un qui en utilise 12 pour traiter quatre files d'attente de réception (ce qui fait un total de cinq unités d'exécution du navigateur pour traiter les cinq files d'attente de réception).

$$n \geq 2 + (3 \times 2) + (5 + 15)$$

Dans ce scénario, vous définirez MAXAPPLS à au moins 28.

2. **Facultatif** : Si vous voulez que le programme Q Apply charge automatiquement les cibles en utilisant l'option LOAD from CURSOR de l'utilitaire LOAD, exécutez la commande suivante : `update dbm cfg using FEDERATED YES.`
3. **Facultatif** : Utilisez la commande de configuration de base de données de mise à jour pour augmenter les valeurs de la base de données cible selon vos besoins en réplication. Les valeurs suivantes de configuration de base de données sont adaptées à de nombreux scénarios de réplication (si la base de données est configurée pour gérer d'autres applications, vos valeurs risquent d'être supérieures à celles-ci) : APPLHEAPSZ 4096, LOGFILSIZ 2000, LOGPRIMARY 20, LOCKLIST 200, DBHEAP 1000, STMTHEAP 4096, LOCKTIMEOUT 30. Pour le paramètre LOGSECOND, la valeur 20 est adaptée à la plupart des scénarios. Si vous comptez répliquer de grandes transactions, il est recommandé de définir LOGSECOND = -1 sur DB2 Version 8.1 ou ultérieure pour éviter la plupart des problèmes d'espace de consignment. Pour LOGBUFSZ, une valeur comprise entre 64 et 512 est recommandée.

Facultatif : liaison des packages de programmes (Linux, UNIX, Windows)

Sous Linux, UNIX et Windows, les packages de programmes sont liées automatiquement la première fois que le programme Q Capture, le programme Q Apply ou le moniteur d'alertes de réplication se connecte à une base de données. Vous pouvez lier les packages manuellement pour spécifier les options de définition d'accès ou les lier au moment où vous prévoyez le moins de conflits dans la base de données.

Facultatif : liaison des packages du programme Q Capture (Linux, UNIX et Windows)

Sous Linux, UNIX et Windows, les packages du programme Q Capture sont liés automatiquement la première fois que le programme se connecte au serveur Q Capture. Vous pouvez choisir de définir les options de définition d'accès ou de lier les packages manuellement au moment où vous prévoyez le moins de conflits dans cette base de données. Cette procédure décrit la manière de lier manuellement les packages du programme Q Capture.

Procédure

Pour lier les packages du programme Q Capture :

1. Connectez-vous au serveur Q Capture en entrant la commande suivante :
`db2 connect to base de données`

où *base de données* est le serveur Q Capture.

2. Accédez au répertoire où se trouvent les fichiers de liens Q Capture.
 - **Linux, UNIX** : `rép_principale_db2/SQLLIB/bnd`, où *rép_principale_db2* est le répertoire principal de l'instance DB2
 - **Windows** : `unité_install_DB2:\...\SQLLIB\bnd`

3. Créez et liez le package Q Capture à la base de données en entrant les commandes suivantes :

```
db2 bind @qcapture.lst isolation ur blocking all
```

où *ur* définit la liste sous le format de lecture non validée pour de meilleures performances.

Cette commande crée des packages dont les noms sont dans le fichier `qcapture.lst`.

Facultatif : liaison des packages de programme Q Apply (Linux, UNIX et Windows)

Sous Linux, UNIX et Windows, les packages de programme Q Apply sont liés automatiquement la première fois que le programme se connecte à la base de données cible et à la base de données source dans le cas où le programme Q Apply gère le chargement de la table cible. Vous pouvez procéder à la liaison manuellement pour définir les options de définition d'accès ou alors au moment où vous prévoyez le moins de conflits dans ces bases de données.

Procédure

Pour lier les packages du programme Q Apply :

1. Accédez au répertoire où se trouvent les fichiers de liens Q Apply :
 - **Linux UNIX** : `rép_principale_db2/SQLLIB/bnd`, où *rép_principale_db2* est le répertoire principal de l'instance DB2
 - **Windows** : `unité_installation_DB2:\...\SQLLIB\bnd`

2. Pour les bases de données source et cible, effectuez les opérations suivantes :

- a. Connectez-vous à la base de données en entrant la commande suivante :
`db2 connect to base de données`

Où *base de données* est le serveur Q Apply ou le serveur Q Capture.

Si la base de données est cataloguée comme base de données éloignée, vous risquez de devoir définir un ID utilisateur et un mot de passe avec la commande **db2 connect to**.

- b. Créez et liez le package de programme Q Apply vers la base de données en entrant les commandes suivantes :

```
db2 bind @qapply.lst isolation ur blocking all grant public
```

où *ur* définit la liste sous le format de lecture non validée.

- c. Facultatif : Si vous prévoyez d'utiliser l'utilitaire DB2 EXPORT pour charger des tables cible à partir d'une source DB2, version 9.7 ou ultérieure et que l'ID utilisateur qui démarre le programme Q Apply ne dispose pas des droits BINDADD, vous devez effectuer la liaison suivante avant le démarrage de Q Apply :

```
db2 bind @db2ubind.lst CONCURRENTACCESSRESOLUTION WAIT_FOR_OUTCOME  
COLLECTION ASN
```

Requis pour les cibles Sybase : Liez manuellement les modules du programme Q Apply à la base de données fédérée en utilisant le niveau d'isolement de la lecture non reproductible (CS) :

```
db2 bind @qapply.lst isolation CS blocking all grant public
```

Facultatif : liaison des packages du moniteur d'alertes de réplication (Linux, UNIX et Windows)

Sous Linux, UNIX et Windows, les packages du moniteur d'alertes de réplication sont liés automatiquement la première fois que le programme se connecte au serveur de contrôle du moniteur ou à tout serveur Q Capture ou Q Apply que vous avez choisi de contrôler. Vous pouvez procéder à une liaison manuelle pour définir les options de définition d'accès ou pour lier les packages manuellement au moment où vous prévoyez le moins de conflits dans ces bases de données.

Procédure

Pour lier les packages de moniteur d'alertes de réplication :

1. Accédez au répertoire où se trouvent les fichiers de liens du moniteur d'alertes de réplication :

- **Linux, UNIX :** *rép_principale_db2/SQLLIB/bnd*, où *rép_principale_db2* est le répertoire principal de l'instance DB2
- **Windows :** *unité_install_DB2:\...\SQLLIB\bnd*

2. Pour chaque serveur de contrôle du moniteur, effectuez la procédure suivante :

- a. Connectez-vous à la base de données en entrant la commande suivante :

```
db2 connect to base de données
```

où *base de données* est le serveur Q Apply ou le serveur Q Capture.

Si la base de données est cataloguée comme base de données éloignée, vous risquez de devoir définir un ID utilisateur et un mot de passe avec la commande **db2 connect to**.

- b. Créez et liez le package du moniteur d'alertes de réplication vers la base de données en entrant la commande suivante :

```
db2 bind @asnmoncs.lst isolation cs blocking all grant public  
db2 bind @asnmonur.lst isolation ur blocking all grant public
```

où *cs* définit la liste sous le format de lecture non reproductible et *ur* définit la liste sous le format de lecture non validée.

Ces commandes créent des packages dont les noms se trouvent dans les fichiers *asnmoncs.lst* et *asnmonur.lst*.

3. Connectez-vous à chaque serveur que vous contrôlez et auquel le moniteur d'alertes de réplication est connecté puis créez et liez le package de moniteur d'alertes de réplication à la base de données en entrant la commande suivante :

```
db2 bind @asnmonit.lst isolation ur blocking all grant public
```

où ur définit la liste sous le format de lecture non validée.

Cette commande crée des packages dont les noms sont dans le fichier `asnmonit.lst`.

Création de tables de contrôle pour les programmes Q Capture et Q Apply

Avant de publier ou de répliquer des données, vous devez créer des *tables de contrôle* pour un programme Q Capture, un programme Q Apply ou les deux. Les tables de contrôle stockent des informations sur les abonnements Q, les publications, les files d'attente de messages, les paramètres opérationnels et les préférences utilisateurs.

Avant de commencer

- Le programme de ligne de commande ASNCLP ou le Centre de réplication doit être capable de se connecter au serveur sur lequel vous souhaitez créer les tables de contrôle :
 - Si vous répliquez de DB2 vers DB2, les outils d'administration doivent alors être capables de se connecter au serveur source et cible.
 - Si vous répliquez de DB2 vers un serveur non-DB2, les outils doivent alors être capables de se connecter au serveur source, au serveur fédéré DB2 et au serveur cible non-DB2.
- Vous devez posséder les noms des objets WebSphere MQ suivants :

Tables de contrôle Q Capture

- Un gestionnaire de files d'attente avec lequel travaille le programme Q Capture.
- Une file d'attente locale à utiliser comme file d'attente d'administration.
- Une file d'attente locale à utiliser comme file d'attente de redémarrage.

Tables de contrôle Q Apply

- Un gestionnaire de files d'attente avec lequel travaille le programme Q Apply.

Conseil : Utilisez la commande **VALIDATE WEBSHERE MQ ENVIRONMENT FOR** dans le ASNCLP, ou la commande `validate controls` dans le Centre de réplication pour vous assurer que les gestionnaires de files d'attente et les files d'attente que vous avez spécifiés pour les tables de contrôle existent et possèdent des propriétés correctes.

Restrictions

Pour les bases de données partitionnées, tous les espaces table utilisés par les tables de contrôle doivent figurer dans un espace table à partition unique se trouvant dans la partition de catalogue.

A propos de cette tâche

Chaque instance du programme Q Capture ou du programme Q Apply a son propre ensemble de tables de contrôle, identifiées par le schéma Q Capture ou le

schéma Q Apply. Par exemple, la table de contrôle stockant les paramètres opérationnels d'un programme Q Capture avec un schéma de ASN1 serait nommée ASN1.IBMQREP_CAPPARMS.

Par défaut, les tables de contrôle sont situées dans deux espaces table sur z/OS, l'un pour les tables nécessitant un verrouillage au niveau de la page et l'autre pour les tables nécessitant un verrouillage au niveau des lignes. Les tables de contrôle sont créées par défaut dans un espace table sur Linux, UNIX et Windows. Vous pouvez personnaliser l'emplacement de la création de chaque table de contrôle et vous pouvez également spécifier les espaces table existants ou en créer de nouveaux.

Cibles fédérées : si vous répliquez vers une cible non-DB2, voir «Création de tables de contrôle Q Apply pour la réplication Q fédérée», à la page 156.

En ce qui concerne la réplication entre homologues, le programme Q Capture et le programme Q Apply fonctionnent comme une paire sur chaque serveur. Les deux ensembles de tables de contrôle doivent avoir le même schéma.

Procédure

Pour créer des tables de contrôle Q Capture ou Q Apply, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande CREATE CONTROL TABLES FOR. Par exemple, les commandes suivantes définissent l'environnement et créent des tables de contrôle dans la base de données SAMPLE avec un schéma Q Capture de ASN1 :</p> <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET QMANAGER "QM1" FOR CAPTURE SCHEMA; SET CAPTURE SCHEMA SOURCE ASN1; SET RUN SCRIPT LATER;</pre> <p>CREATE CONTROL TABLES FOR CAPTURE SERVER USING RESTARTQ "ASN1.QM1.RESTARTQ" ADMINQ "ASN1.QM1.ADMINQ" MEMORY LIMIT 64 MONITOR INTERVAL 600000;</p> <p>La commande CREATE spécifie une file d'attente de redémarrage et une file d'attente d'administration, elle double la taille par défaut de mémoire disponible pour construire des transactions à 64 Mo, et réduit à 600000 millisecondes (une minute) la fréquence par défaut d'enregistrement des informations sur la performance.</p>
Centre de réplication	<p>Utilise l'assistant Créer des tables de contrôle. Pour ouvrir les assistants, cliquez avec le bouton droit de la souris sur le dossier Serveurs Q Capture puis cliquez sur Créer des tables de contrôle Q Capture, ou cliquez avec le bouton droit de la souris sur le dossier Serveurs Q Apply puis cliquez sur Créer des tables de contrôle Q Apply.</p>

Création de tables de contrôle sur une version différente (Linux, UNIX, Windows)

Vous pouvez utiliser une version de réplication ou de publication d'événements qui est plus récente que votre version de DB2. Pour ce faire, vous devez créer des tables de contrôle de réplication qui correspondent à la version de réplication et de publication d'événements au lieu de la version de DB2. Par exemple, si vous envisagez d'utiliser la réplication, niveau de version 9.7 avec une base de données DB2, version 9.5, vos tables de contrôle doivent se situer au niveau de version 9.7.

Avant de commencer

Pour créer des tables de contrôle sur une version différente, vous devez tout d'abord installer le produit de réplication ou de publication sans mettre à niveau DB2.

A propos de cette tâche

Le diagramme suivant affiche une configuration dans laquelle le Centre de réplication crée des tables de contrôle Q Apply, version 9.7 sur une base de données DB2 version 9.1. Vous pouvez également utiliser le programme de ligne de commande ASNCLP pour spécifier une version différente pour les tables de contrôle.

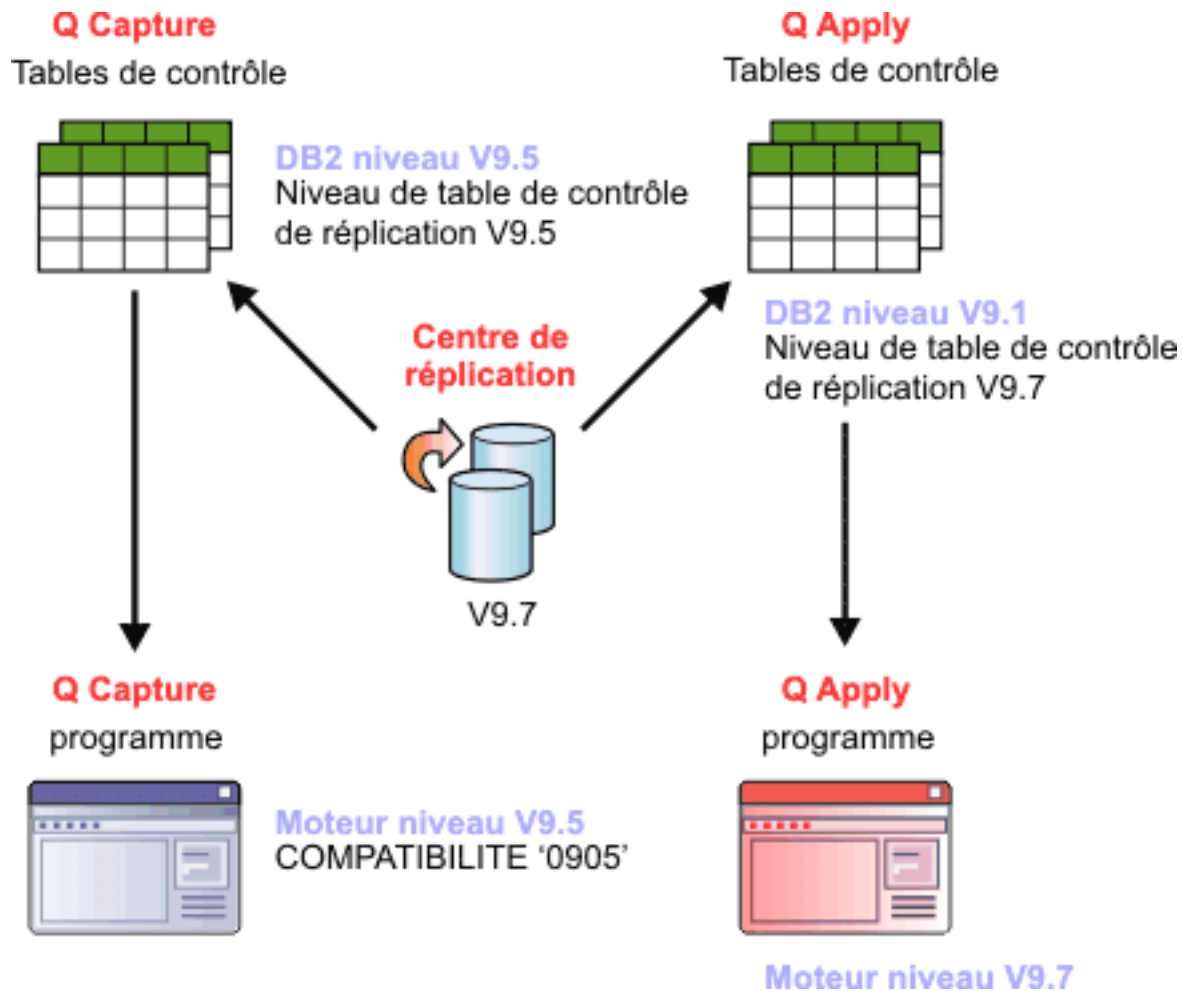


Figure 8. Création de tables de contrôle de réplication avec une version différente de celle de DB2. Centre de réplication, version 9.7 peut créer des tables de contrôle sur les versions V9.7, V9.5 ou V9.1. Dans le diagramme, le Centre de réplication crée des tables de contrôle V9.5 par défaut sur une base de données DB2 V9.5 et crée des tables de contrôle V9.7 sur une base de données DB2 V9.1. Ainsi, le niveau du programme Q Capture est V9.5 et le niveau du programme Q Apply est V9.7. Toutefois, Q Capture étant à un niveau de version inférieur à Q Apply, la compatibilité pour Q Capture doit être définie sur 0905. Ainsi, Q Capture envoie des messages V9.5 à Q Apply et les fonctions V9.5 sont prises en charge.

Remarque : Lorsque les tables de contrôle Q Capture et Q Apply se trouvent à des niveaux de version différents, la valeur de la colonne COMPATIBILITY de la table de contrôle IBMQREP_CAPPARMS détermine le niveau des messages envoyés par Q Capture. Le niveau de compatibilité du serveur Q Capture doit être inférieur ou égal au niveau d'architecture des tables de contrôle du programme Q Apply utilisé. Dans le diagramme, la compatibilité Q Capture est définie sur 0905 car le programme Q Capture et ses tables de contrôle se situent au niveau de version 9.5. Dans ce cas, le programme Q Capture envoie des messages de version 9.5 et même si Q Apply se situe au niveau de version 9.7, les deux programmes utilisent les fonctionnalités de la version 9.5.

Restrictions

- Le produit de réplication et de publication doit se situer à un niveau de version identique ou ultérieur à celui de DB2. DB2 doit être à un niveau de version 8.2 ou supérieur.
- Le niveau de version des tables de contrôle doit correspondre au niveau de version du produit de publication ou de réplication.

- La spécification de la version des tables de contrôle n'est pas prise en charge sous DB2 for z/OS. Sous z/OS, le Centre de réplication crée toujours des tables de contrôle qui correspondent à la version du client DB2 sous laquelle le Centre de réplication fonctionne. Vous pouvez également l'exemple SQL fourni avec les produits de réplication et de publication sous z/OS pour créer des tables de contrôle.
- Certaines fonctions de réplication et de publication dépendent du niveau de DB2. Par exemple, vous pouvez effectuer une réplication uniquement à partir de tables compressées ou de tables partitionnées par intervalle si la source DB2 est au niveau de version 9.7 ou supérieur. DB2 décompresse les enregistrements des journaux avant de les transmettre au programme Q Capture. Les données décompressées sont transmises de la source vers la cible sur laquelle la compression n'est pas nécessaire.

Procédure

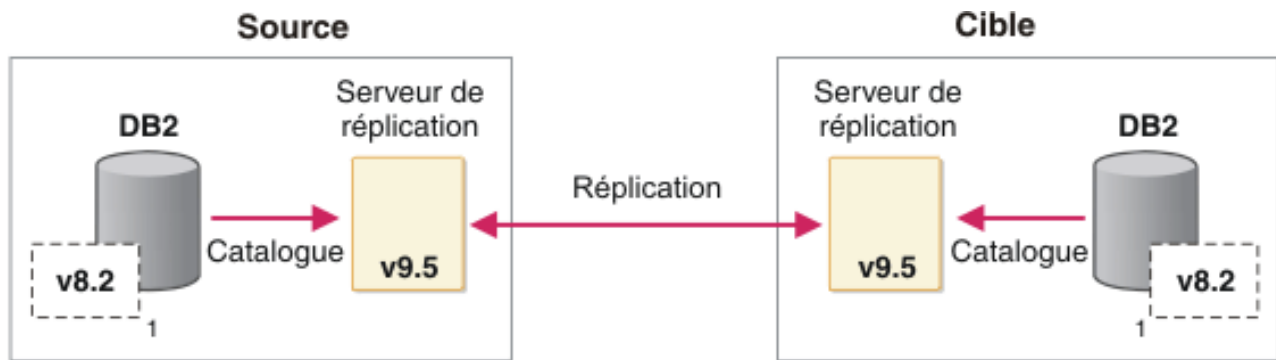
Pour créer des tables de contrôle sur une version différente, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Dans la commande CREATE CONTROL TABLES FOR, utilisez le mot clé de version pour spécifier le niveau de version, comme indiqué dans l'exemple suivant : CREATE CONTROL TABLES FOR CAPTURE SERVER USING RESTARTQ "ASN1.QM1.RESTARTQ" ADMINQ "ASN1.QM1.ADMINQ" version 9.5
Centre de réplication	Sur la page de synthèse de l'assistant Créer des tables de contrôle Q Capture ou Créer des tables de contrôle Q Apply, cliquez sur Modifier en regard de la liste relative à la version de Q Capture ou Q Apply. Dans la fenêtre Modifier la version de la table de contrôle, spécifiez la version.

Configuration pour les bases de données DB2 distantes ou antérieures

Vous pouvez configurer InfoSphere Replication Server ou Data Event Publisher avec des bases de données DB2 antérieures afin de pouvoir utiliser les fonctionnalités de réplication les plus récentes sans devoir migrer ces bases de données existantes. Vous pouvez également configurer InfoSphere Replication Server ou Data Event Publisher sur un autre serveur que ceux hébergeant vos bases de données DB2. Les deux configurations utilisent des étapes similaires.

La figure suivante illustre la configuration de base pour utiliser InfoSphere Replication Server avec une version antérieure de DB2. Dans cet exemple, il s'agit de InfoSphere Replication Server version 9.5 avec DB2 version 8.2.



(1) Cette configuration est actuellement valide avec les versions DB2, à partir de la version 8.2

Figure 9. Configuration de Replication Server avec des bases de données DB2 antérieures

La figure suivante illustre la configuration de base pour utiliser des bases de données DB2 distantes.

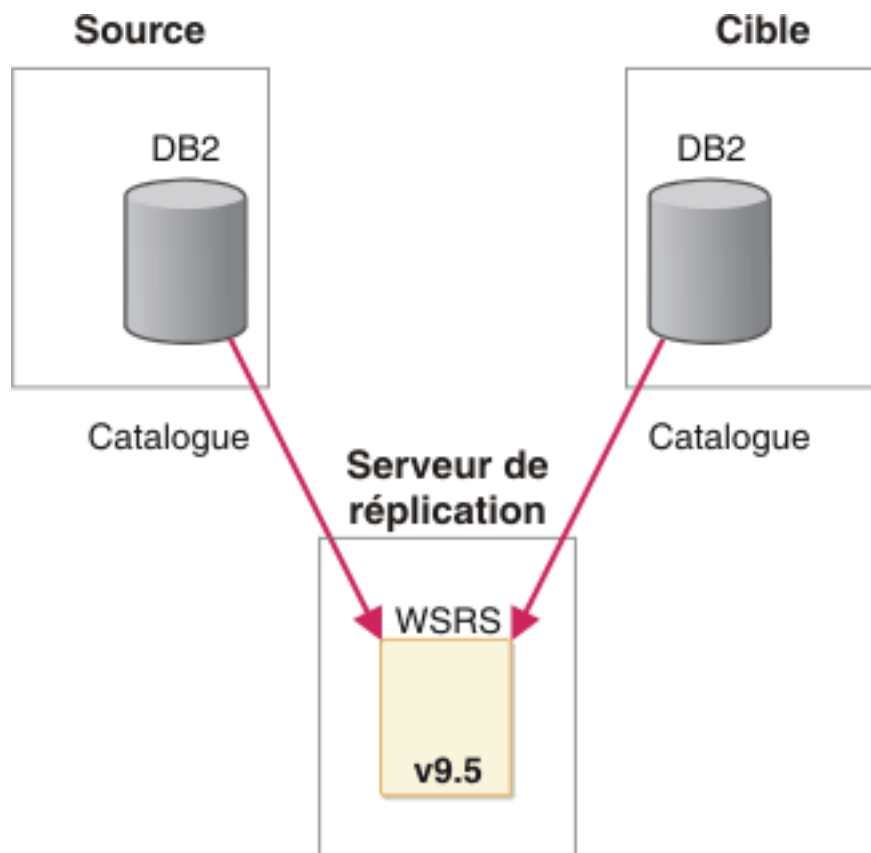


Figure 10. Configuration avec des bases de données DB2 distantes

Avant de commencer

Si vous prévoyez d'utiliser une nouvelle version de Replication Server ou de Data Event Publisher sur le même système qu'une base de données DB2 antérieure, voir Scénarios d'installation pour connaître les informations importantes à prendre en compte pour l'installation.

Si vous prévoyez d'utiliser un programme Q Capture avec une version ou une édition différente de celle votre programme Q Apply, voir Prise en charge de la coexistence de la réplication Q et de la publication d'événements version 9.7

A propos de cette tâche

L'installation de Replication Server ou de Data Event Publisher ne met pas à niveau vos bases de données DB2 existantes, même si la version de DB2 est antérieure. Après installation et acceptation de la licence de InfoSphere Replication Server ou de Data Event Publisher, les programmes de réplication pourront interagir avec toute base de données DB2 de niveau d'édition égale ou inférieure (jusqu'à la version 8.2), que cette base de données réside sur une autre instance DB2 ou sur un autre système. Dans le cadre de cette installation, une nouvelle instance DB2 est créée, mais cette instance DB2 n'a aucun besoin d'être démarrée et il est inutile d'y créer une base de données.

Les configurations à versions mixtes ou réplication à distance nécessitent des étapes supplémentaires :

- Les bases de données ou noeuds DB2 antérieurs ou distants doivent être catalogués sur l'instance contenant les programmes de réplication.
- Dans les configurations à versions mixtes, les tables de contrôle de réplication doivent être créées pour correspondre au niveau de version plus élevé des programmes de réplication.
- Dans les configurations à versions mixtes, le propriétaire de l'instance DB2 de plus haut niveau installée avec les tout derniers programmes de réplication doit disposer des droits en lecture sur les journaux DB2 de la version antérieure de DB2.

Restrictions

Les outils d'administration de réplication ne créent pas de tables de contrôle pré-version 10.1 pour une base de données source ou cible sous DB2 for Linux, UNIX et Windows version 10.1.

Pour Q Capture ou Q Apply qui sont locaux à DB2

- La réplication depuis des tables compressées ou partitionnées sur DB2 pour Linux, UNIX et Windows est uniquement prise en charge pour DB2 version 9.7 ou ultérieure. DB2 décompresse les enregistrements des journaux avant de les transmettre au programme Q Capture. Les données décompressées sont transmises de la source vers la cible sur laquelle la compression n'est pas nécessaire.
- Pour répliquer des opérations de chargement, la base de données source et le programme Q Capture doivent avoir le correctif APAR PK78558 ou un niveau de correctif plus récent sous z/OS ou ils doivent posséder la version 9.7 ou ultérieure sous Linux, UNIX et Windows.

Pour Q Capture ou Q Apply qui sont distants de DB2

- Le programme Q Capture, le programme Q Apply et les bases de données source et cible doivent être sur la même plateforme Linux, UNIX ou Windows et être dotés de la même architecture binaire, par exemple 32/32 ou 64/64 bits.
- Vous ne pouvez pas utiliser les outils d'administration de réplication pour répertorier ou valider des files d'attente.

- La valeur endian et la page de codes du système sur lequel sont exécutés les programmes Q Capture doivent correspondre à la valeur endian et à la page de codes de la base de données source et au système d'exploitation.

Procédure

1. Installez Replication Server ou Data Event Publisher et créez une instance comme vous y invite le programme d'installation. Vous pouvez également installer les produits sur le même système qu'une version antérieure de DB2 (jusqu'à la version j8.2) ou bien les installer sur un système distant.
2. Cataloguez le noeud et la base de données DB2 source ou cible sur l'instance nouvellement installée à l'aide des commandes **CATALOG NODE** et **CATALOG DATABASE**. Ouvrez une fenêtre de commande DB2 à partir de l'instance DB2 utilisée par la réplication ou par la publication d'événements et cataloguez le noeud et la base de données de la base de données distante ou antérieure.
3. Créez un fichier de mot de passe asnpwd à utiliser par les programmes de réplication et de publication d'événements.
4. Si la base de données source ou cible est une version antérieure à celle utilisée par Q Capture ou Q Apply, utilisez ASNCLP ou le Centre de réplication pour créer des tables de contrôle qui correspondent à la version des programmes de réplication.
5. Utilisez le Centre de réplication ou ASNCLP pour créer les mappes de files d'attente et les abonnements Q.
6. Si la base de données doit être une source de réplication, accordez les droits d'accès SYSADM ou DBADM à l'utilisateur que le programme Q Capture utilisera pour la connexion à la source. Le programme Q Capture pourra ainsi extraire les enregistrements du journal DB2.
7. Démarrez Q Capture et Q Apply à partir de l'instance de réplication, qu'il s'agisse d'une version nouvelle ou de la même version sur un système distant. Par exemple, démarrez la réplication à partir de l'instance nouvellement installée en fournissant l'alias de la base de données DB2 antérieure ou distante spécifiée dans la commande **CATALOG DATABASE**.

Chapitre 5. Configuration de la réplication Q unidirectionnelle

Avec la réplication unidirectionnelle, vous pouvez répliquer les données dans un sens, depuis une table source vers une table cible, ou manipuler les données dans la table cible à l'aide de procédures stockées. L'abonnement Q peut avoir pour cible un serveur DB2 ou un serveur non DB2.

Le programme Q Capture réplique les transactions depuis une table source puis les place dans une file d'attente d'envoi dans un format compressé. Le programme Q Apply reçoit ensuite ces messages compressés depuis une file d'attente de réception et applique les transactions à une table cible ou les envoie à une procédure stockée.

Réplication unidirectionnelle

La réplication unidirectionnelle permet de répliquer des données de tables sources vers des tables cible ou des procédures stockées.

La réplication unidirectionnelle est une configuration de réplication Q ayant les caractéristiques suivantes :

- Les transactions effectuées dans une table source sont répliquées sur les files d'attente de WebSphere MQ vers une table cible ou envoyées en tant que paramètres d'entrée à une procédure stockée en vue de manipuler les données.
- Les transactions qui ont lieu sur la table cible ne sont pas répliquées en retour sur la table source.
- La table cible est généralement en lecture seule et n'est pas mise à jour par d'autres applications que le programme Q Apply.

Le programme Q Capture réplique les transactions depuis une table source puis les place dans une file d'attente d'envoi dans un format condensé. Le programme Q Apply reçoit ensuite ces messages condensés depuis une file d'attente de réception et applique les transactions à une table cible (une table DB2 ou un pseudonyme sur un serveur DB2 fédéré) ou les transmet à une procédure stockée.

A partir de n'importe quelle table source, vous pouvez répliquer toutes les colonnes et lignes ou uniquement un sous-ensemble de colonnes et de lignes. Si vous souhaitez transformer les données, vous pouvez spécifier un programme Q Apply pour qu'il transmette les transactions d'une table source comme paramètre d'entrée à une procédure stockée que vous fournissez. La procédure stockée peut mettre à jour les données sur un serveur DB2 ou un serveur non-DB2.

Dans une réplication unidirectionnelle, les objets suivants existent entre les serveurs :

Mappes de files de réplication

Vous devez créer au moins une mappe de files de réplication pour transporter les données depuis le programme Q Capture sur le serveur source vers le programme Q Apply sur le serveur cible (ou le serveur fédéré DB2 si vous les répliquez vers une table cible non DB2).

Abonnements Q

Il existe un abonnement Q pour chaque couple table source-table cible ou chaque couple table source-procédure stockée. Par exemple, si vous

disposez d'une table source sur le serveur SERVER_RED, d'une table cible sur le serveur SERVER_GREEN et d'une autre table cible sur le serveur SERVER_BLUE, il y a alors deux abonnements Q :

- Un abonnement de la table source sur le serveur SERVER_RED vers la table cible sur le serveur SERVER_GREEN
- Un abonnement de la table source sur SERVER_RED vers la table cible sur SERVER_BLUE

La figure 11 montre le résultat obtenu lorsque vous mappez trois tables source vers trois tables cible, en une fois, pour une réplication unidirectionnelle. Dans cet exemple, il y a trois abonnements Q distincts. Les modifications de la source A sont répliquées vers la cible A, les modifications de la source B sont répliquées vers la cible B, etc. Les modifications de la source A ne peuvent pas être répliquées vers la cible B. Ces couples source et cible utilisent la même file d'attente de réplication et les mêmes programmes Q et Apply.

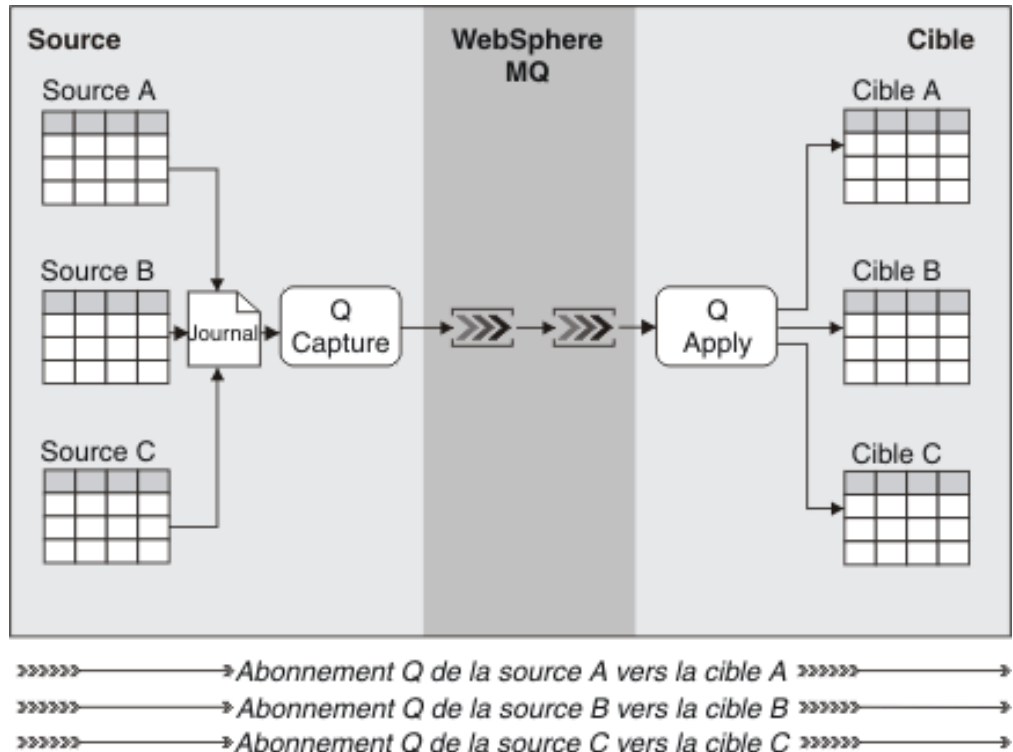


Figure 11. Plusieurs abonnements Q destinés à la réplication unidirectionnelle. Dans la réplication unidirectionnelle, les modifications de chaque table source sont répliquées sur les files d'attente de WebSphere MQ vers une table cible spécifique.

Regroupement de mappes de file de réplication et d'abonnements Q

Avant de définir les abonnements Q et les mappes de files de réplication, vous devez tout d'abord planifier la manière dont vous voulez les regrouper.

Chaque abonnement Q relie une table source unique à une table cible unique ou à une procédure stockée. Lorsque vous définissez un abonnement Q, vous devez également définir quelle mappe de files de réplication est utilisé pour transporter les données depuis la table source jusqu'à la table cible ou la procédure stockée.

Entre autres, chaque mappe de file d'attente de réplication identifie la file d'attente WebSphere MQ vers laquelle le programme Q Capture envoie les modifications et la file d'attente WebSphere MQ à partir de laquelle le programme Q Apply reçoit ces dernières avant de les appliquer à la table cible ou de les transmettre à une procédure stockée. Il est possible d'utiliser une mappe de file d'attente de réplication unique pour plusieurs abonnements Q. Dans ce cas, vous devez décider quels abonnements Q utilisent la même mappe de file d'attente de réplication pour transférer les données.

Si vous décidez de regrouper des abonnements Q et des mappes de files de réplication, conformez-vous aux règles suivantes :

- Une file d'attente de WebSphere MQ ne peut pas être partagée par plusieurs programmes Q Capture ou plusieurs programmes Q Apply.
- Un seul programme Q Capture ou Q Apply peut par contre écrire sur plusieurs files d'attente ou lire à partir de plusieurs files. Un programme Q Capture peut par exemple écrire des données sur plusieurs files d'attente d'envoi et un programme Q Apply peut lire et appliquer des données à partir de plusieurs file d'attente de réception.
- Vous pouvez créer plus d'une mappe de files de réplication entre n'importe quelle couple de programmes Q Capture-Q Apply. Chaque programme Q capture et Q Apply peut également fonctionner avec plusieurs mappes de files de réplication. Par exemple, un programme Q Capture peut envoyer des messages à plusieurs files d'attente d'envoi et un programme Q Apply peut extraire les messages de plusieurs files d'attente de réception.

Fonctionnement de Q Capture avec la file d'attente d'envoi

Pour une mappe de file de réplication, Q Capture enregistre les modifications pour toutes les tables pour lesquelles il existe un abonnement Q actif. Q Capture stocke en mémoire ces modifications jusqu'à ce qu'il lise l'entrée de validation ou d'abandon correspondante du journal de base de données. Q Capture envoie alors les informations concernant les transactions effectuées à toutes les files d'attente d'envoi qui ont été définies pour les abonnements Q.

Fonctionnement de Q Apply avec la file d'attente de réception

Q Apply lance une unité d'exécution du navigateur pour chaque file d'attente de réception ayant été définie pour un schéma Q Apply donné. Pour chaque navigateur, les messages de transactions que l'unité d'exécution du navigateur de Q Apply lit à partir de la file d'attente de réception sont appliqués par au moins un agent Q Apply et au plus par le nombre maximum d'agents que vous avez défini. Dans le contexte d'une file d'attente de réception, les transactions seront exécutées en série, au niveau des dépendances entre transactions, en se basant sur les relations entre les contraintes d'unicité ou les clés externes. Si aucune dépendance de contrainte n'existe entre les transactions, celles-ci sont exécutées autant que possible en parallèle.

Suggestions pour regrouper des abonnements Q similaires avec des mappes de files de réplication

En règle générale, vous devez créer des abonnements Q pour les tables qui sont utilisées pour des transactions avec au moins une application, afin que ces tables partagent une mappe de files de réplication commune. Le regroupement d'abonnements Q similaires avec la même mappe de files de réplication garantit la cohérence des transactions des données lorsque Q Apply applique ces données aux

tables cible ou aux procédures stockées. Q Apply appliquant déjà des modifications pour ces transactions en parallèle, il n'est pas nécessaire de créer plusieurs mappes de files de réplication pour atteindre un degré de parallélisme élevé dans l'application des données.

Si vous définissez des abonnements Q servant pour des transactions liées, en vue d'envoyer des données via des mappes de files de réplication indépendants, Q Capture divise les données entre les diverses files d'attente d'envoi. Plusieurs navigateurs de Q Apply liés aux files d'attente de réception appliquent les données de manière indépendante.

Les abonnements Q qui possèdent des dépendances doivent partager la même mappe de files de réplication. Le navigateur de Q Apply détecte au niveau de la file d'attente de réception les dépendances entre les transactions, de telle manière que tous les abonnements Q impliquant des tables dépendantes puissent utiliser la même file d'attente de réception. Si les transactions dépendantes sont envoyées à des files d'attente de réception différentes via des mappes de files de réplication différentes, il se peut qu'il n'y ait pas de cohérence en matière de transactions entre la base de données cible et la base de données source.

Si plusieurs applications mettent à jour le serveur source mais ne mettent pas à jour les mêmes tables, et si vous configurez une paire unique de programmes Q Capture et Q Apply pour qu'ils répliquent les données du serveur source vers le serveur cible, vous devez alors penser à définir plusieurs mappes de file d'attente de réplication pour cette paire de programmes Q Capture et Q Apply à utiliser. Tous les abonnements Q associés dans des transactions pour chaque application sont ensuite répliqués sur l'une de ces mappes de file d'attente de réplication. Une telle configuration peut avoir des avantages comme l'isolement des incidents ou l'accroissement du rendement. Il est possible d'obtenir un rendement encore plus élevé et un meilleur isolement en cas d'incident en configurant plusieurs couples de programmes Q Capture - Q Apply, avec leur propre mappe de file d'attente de réplication. Cependant, ces avantages sont contrebalancés par une consommation plus importante de l'unité centrale et un environnement de réplication plus complexe.

Création de mappes de files de réplication

Lorsque vous créez des abonnements Q, vous spécifiez les files d'attente WebSphere MQ vers lesquelles envoyer les données en associant chaque abonnement Q à une mappe de files de réplication. Vous pouvez créer une mappe de files de réplication avant de commencer à créer des abonnements Q ou comme une des étapes de création d'abonnements Q.

Avant de commencer

- Planifiez la manière dont vous voulez regrouper les mappes de files de réplication et les abonnements Q.
- Sur le serveur contenant les tables source des abonnements Q, créez les tables de contrôle du programme Q Capture.
- Sur le serveur contenant les tables cible des abonnements Q, créez les tables de contrôle du programme Q Apply.
- Assurez-vous que vous avez défini les objets appropriés dans WebSphere MQ.

Restrictions

La même file d'attente d'envoi ne peut pas être utilisée à la fois pour une réplication Q et pour une publication d'événements car une file d'attente d'envoi peut transférer des messages condensés (pour une réplication Q) ou des messages XML (pour la publication d'événements), mais pas les deux.

Procédure

Pour créer une mappe de files de réplication, appliquez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande CREATE REPLQMAP. Par exemple, les commandes suivantes définissent l'environnement et créent une mappe de files de réplication SAMPLE_ASN1_TO_TARGET_ASN1 : ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE ASN1; SET SERVER TARGET TO DB TARGET; SET APPLY SCHEMA ASN1; SET RUN SCRIPT LATER; CREATE REPLQMAP SAMPLE_ASN1_TO_TARGET_ASN1 USING ADMINQ "ASN1.QM1.ADMINQ" RECVQ "ASN1.QM1_TO_QM2.DATAQ" SENDQ "ASN1.QM1_TO_QM2.DATAQ" NUM APPLY AGENTS 8 HEARTBEAT INTERVAL 5;
Centre de réplication	Utilisez la fenêtre Créer une mappe de files de réplication. Pour ouvrir la fenêtre depuis la page Serveurs de l'assistant Création d'abonnements Q ou depuis l'arborescence des objets, développez le schéma Q Capture ou Q Apply approprié, cliquez avec le bouton droit sur le dossier Mappes de files de réplication et sélectionnez Créer .

Conseil : Vous pouvez utiliser l'outil d'administration de réplication pour valider la file d'attente d'envoi, de réception et la file d'attente d'administration que vous spécifiez pour une mappe de files de réplication et pour envoyer des messages de test entre les files d'attente. Pour valider, utilisez la commande VALIDATE WSMQ ENVIRONMENT FOR dans ASNCLP ou cliquez sur **Valider les files d'attente** dans la fenêtre Créer une mappe de files de réplication du Centre de réplication. Pour envoyer des messages de test, utilisez la commande VALIDATE WSMQ MESSAGE FLOW FOR REPLQMAP dans ASNCLP ou la fenêtre Valider les files d'attente WebSphere MQ du Centre de réplication. Lorsque vous créez une mappe de files de réplication, vous pouvez spécifier les options suivantes :

File d'attente d'envoi

File d'attente WebSphere MQ où le programme Q Capture envoie des transactions source et des messages d'information. Lorsque vous définissez une mappe de files de réplication, vous devez sélectionner une file d'attente d'envoi configurée pour transférer des messages condensés.

File d'attente de réception

File d'attente WebSphere MQ à partir de laquelle le programme Q Apply reçoit des transactions source et des messages d'information.

File d'attente d'administration

File d'attente que le programme Q Apply utilise pour envoyer des

messages de contrôle au programme Q Capture. Les messages que le programme Q Apply envoie sur cette file d'attente ont différents objectifs, comme le fait d'indiquer au programme Q Capture de démarrer l'envoi de messages ou d'initier le processus de chargement pour une table cible.

Longueur maximale du message

Taille maximale (en kilooctets) d'un message que le programme Q Capture peut mettre sur cette file d'attente d'envoi. Cette taille maximale de message doit être inférieure ou égale à l'attribut de taille maximale de message (MAXMSGL) de WebSphere MQ défini pour la file d'attente ou le gestionnaire de file d'attente.

Action d'erreur de file d'attente

Action que le programme Q Capture effectue lorsqu'une file d'attente d'envoi n'accepte plus de messages en raison d'une erreur, lorsque la file d'attente est saturée par exemple :

- Arrêt du fonctionnement
- Arrête de placer des messages dans la file d'attente en erreur mais continue à placer des messages dans les autres files d'attente

Nombre d'agents Q Apply

Nombre d'unités d'exécution ou d'agents que le programme Q Apply utilise pour appliquer des transactions de manière simultanée à partir de cette file d'attente de réception. Pour demander à ce que ces transactions soient appliquées dans l'ordre où elles ont été reçues à partir de la table source, spécifiez un seul agent Q Apply. Pour appliquer parallèlement les modifications au serveur cible, spécifiez plusieurs agents Q Apply.

Utilisation maximale de la mémoire Q Apply

Quantité de mémoire maximale (en mégaoctets) que le programme Q Apply utilise comme mémoire tampon pour les messages provenant de cette file d'attente de réception.

Intervalle des pulsations

Fréquence, en secondes, avec laquelle le programme Q Capture envoie des messages sur cette file d'attente pour indiquer au programme Q Apply que le programme Q Capture est toujours en cours de fonctionnement lorsqu'il n'y a aucune transaction à répliquer. La pulsation est envoyée sur le premier intervalle de validation après l'expiration de l'intervalle des pulsations. La valeur 0 indique au programme Q Capture de ne pas envoyer de messages de pulsation.

Cet intervalle de pulsations est différent du paramètre HBINT (intervalle des pulsations) de WebSphere MQ que vous pouvez définir pour une voie WebSphere MQ.

Création d'abonnements Q pour une réplification unidirectionnelle

La réplification Q permet de définir la réplification de données à partir de tables source vers des tables cible ou de manipuler les données de la cible avec des procédures stockées en créant des abonnements Q. Vous devez créer un abonnement Q pour chaque paire source-cible. L'abonnement Q peut avoir pour cible un serveur DB2 ou un serveur non DB2.

Chaque abonnement Q représente un objet unique qui identifie les informations suivantes :

- La table source à partir de laquelle vous voulez effectuer une réplification des modifications

- La table cible ou la procédure stockée vers laquelle vous voulez effectuer une réplification des modifications
- Les lignes et colonnes de la table source que vous voulez répliquer
- La mappe de files de réplification, qui désigne les files d'attente de WebSphere MQ transportant les informations entre le serveur source et le serveur cible

Vous pouvez créer un ou plusieurs abonnements Q à la fois.

Avertissement : Les abonnements Q sont des objets distincts des publications. Les publications ne publient pas de données dans le programme Q Apply mais vers une application de votre choix. Les abonnements Q servent pour la réplification de données et les publications pour la publication de données. Si vous voulez répliquer les modifications d'une table source et que le programme Q Apply applique ces modifications source à une table cible ou les transmette à une procédure stockée pour une manipulation de données, définissez un abonnement Q et non pas une publication.

Création de profils d'objet cible

Vous pouvez créer des profils dans le Centre de réplification ou dans le programme de ligne de commande ASNCLP pour définir votre propre convention d'attribution de nom par défaut pour les objets créés par les outils d'administration au niveau du serveur cible.

A propos de cette tâche

Vous pouvez créer des profils pour les objets suivants :

Centre de réplification

Tables cible, espaces table (ou leurs équivalents pour les cibles non DB2), index et pseudonymes

ASNCLP

Espaces table ou index

Les outils d'administration appliquent les règles d'attribution de nom contenues dans le profil objet cible pour attribuer un nom aux objets qu'ils créent.

Procédure

Pour créer des profils objet cible, appliquez une des méthodes suivantes :

Méthode	Description
<p>Programme de ligne de commande ASNCLP</p>	<p>Utilisez la commande SET PROFILE pour définir les paramètres personnalisés des espaces table ou index créés par le programme ASNCLP. Par exemple, les commandes suivantes définissent l'environnement et créent un profil TBSPROFILE qui définit les options d'index des tables qui suivent le dispositif de verrouillage de page :</p> <pre>ASNCLP SESSION SET TO Q REPLICATION; SET PROFILE TBSPROFILE FOR OBJECT PAGE LOCK INDEX OPTIONS ZOS DB TARGETDB STOGROUP MYSTOGROUPPRIQTY PERCENT OF SOURCE 70;</pre> <p>Après avoir émis une commande SET PROFILE, vous pouvez associer un profil à une commande de tâche en spécifiant le nom du profil dans la commande de tâche.</p> <p>La portée du profil dure seulement le temps de la session en cours. Lorsque vous quittez la session ASNCLP, les informations relatives au profil ne sont pas sauvegardées pour la session suivante.</p>

Méthode	Description
Centre de réplication	<p>Utilisez le bloc-notes Gestion des profils d'objets cible. Pour ouvrir le bloc-notes, cliquez avec le bouton droit de la souris sur un serveur Q Apply dans l'arborescence des objets et cliquez sur Gestion des profils d'objets cible.</p> <p>Le Centre de réplication possède un profil objets cible par défaut que vous pouvez modifier ou substituer lorsque vous créez de nouveaux abonnements Q. Les objets existants ne sont pas renommés.</p> <p>Le Centre de réplication enregistre un profil objet cible pour chaque serveur cible. Lorsque vous créez un abonnement Q, le Centre de réplication utilise le profil du serveur cible pour déterminer le propriétaire et le nom de la table cible. S'il existe une table avec ce propriétaire et ce nom, le Centre de réplication utilise la table existante comme cible pour l'abonnement Q. S'il n'existe pas de table avec ce propriétaire et ce nom, le Centre de réplication crée une table avec ce propriétaire et ce nom pour vous.</p> <p>La convention d'attribution de nom des objets cible est constituée de trois parties :</p> <ul style="list-style-type: none"> • un préfixe • une base, qui est soit le nom de l'objet de base de données associé, soit un horodatage • un suffixe <p>Vous pouvez aussi spécifier si vous souhaitez créer des tables cible dans des espaces table nouveaux ou existants (ou des espaces de base de données pour certaines cibles non DB2). Vous pouvez définir des paramètres opérationnels de l'espace de table, y compris si l'espace de table cible doit utiliser le même partitionnement que l'espace de table source, si le serveur cible est un sous-système DB2 sous z/OS.</p> <p>Vous pouvez aussi définir les règles de troncature pour les noms de ces objets. Si un nom d'objet qui est le préfixe, la base et le suffixe, dépasse la longueur maximum de votre système d'exploitation, les règles de troncature demandent au centre de réplication de raccourcir la base du nom, soit sur la gauche, soit sur la droite, jusqu'à ce que le nom atteigne la longueur maximum autorisée par votre système d'exploitation.</p>

Création d'abonnements Q pour une réplication unidirectionnelle

En créant des abonnements Q pour une réplication unidirectionnelle, vous définissez la manière dont les données sont répliquées à partir des tables source dans des tables cible ou sont envoyées à des paramètres dans une procédure pour la manipulation des données.

Avant de commencer

- Planifiez la manière dont vous voulez regrouper les mappes de files de réplication et les abonnements Q.
- Créez les tables de contrôle pour le programme Q Capture dans le serveur contenant la table source pour l'abonnement Q.

- Créez les tables de contrôle pour le programme Q Apply dans le serveur contenant la cible pour l'abonnement Q.
- Définissez les files d'attente pour la réplication et leurs attributs en créant une mappe de file de réplication. (Vous pouvez effectuer cette tâche avant de créer un abonnement Q ou pendant que vous créez un abonnement Q.)
- Préparez la procédure stockée si vous souhaitez que le programme Q Apply envoie des modifications source plutôt dans une procédure stockée que dans une table cible.

Restrictions

- Les affichages ne peuvent pas être des sources ou des cibles pour les abonnements Q.
- Les colonnes IDENTITY dans la table cible ne peuvent pas être définies comme GENERATED ALWAYS.

z/OS

- Ne sélectionnez pas les colonnes ROWID pour réplication à moins que la colonne ROWID ne soit le seul index à entrées uniques indiqué pour la réplication. La réplication pour des colonnes ROWID n'est pas prise en charge pour la réplication bidirectionnelle ou entre homologues.

Recommandation : Utilisez plutôt une colonne IDENTITY comme index à entrées uniques pour la réplication.
Si vous répliquez des colonnes LOB, vous devez avoir un index unique en plus de l'index unique ROWID.

Linux UNIX Windows

- La réplication est prise en charge à partir de bases de données à partitions multiples. Le nombre de partitions pris en charge par la réplication n'est pas limité.
- La réplication est prise en charge à partir de tables utilisant la compression de valeur ou la compression de ligne. DB2 décompresse les enregistrements des journaux avant de les transmettre au programme Q Capture. Les données décompressées sont transmises de la source vers la cible sur laquelle la compression n'est pas nécessaire.
- La réplication est prise en charge à partir de tables de requêtes matérialisées (MQT).

Procédure

Pour créer un abonnement Q pour une réplication unidirectionnelle à partir d'une table source dans une table cible ou une procédure stockée, appliquez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande CREATE QSUB. Par exemple, les commandes suivantes définissent l'environnement et créent un abonnement Q pour une réplication unidirectionnelle, EMPLOYEE0001, avec les caractéristiques suivantes :</p> <ul style="list-style-type: none"> • La mappe de files de réplication est SAMPLE_ASN1_TO_TARGETDB_ASN1. • Le programme Q Apply charge les tables cible en utilisant les utilitaires EXPORT et IMPORT. • La colonne EMPNO est utilisée en tant que colonne de clé pour la réplication pour déterminer l'unicité d'une ligne. <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE ASN1; SET SERVER TARGET TO DB TARGET; SET APPLY SCHEMA ASN1; SET RUN SCRIPT LATER;</pre> <pre>CREATE QSUB USING REPLQMAP SAMPLE_ASN1_TO_TARGETDB_ASN1 (SUBNAME EMPLOYEE0001 EMPLOYEE OPTIONS HAS LOAD PHASE I TARGET NAME TGTEMPLOYEE KEYS (EMPNO) LOAD TYPE 2)</pre>
Centre de réplication	<p>Utilisez l'assistant Créer des abonnements Q. Pour ouvrir l'assistant, développez le schéma Q Capture ou Q Apply approprié, cliquez avec le bouton droit de la souris sur le dossier abonnements Q et sélectionnez Créer.</p> <p>Vous pouvez créer un ou plusieurs abonnements Q en utilisant l'assistant.</p> <p>Page des lignes et des colonnes</p> <p>Lorsque vous créez plusieurs abonnements Q en même temps, le Centre de réplication suppose que vous souhaitez répliquer toutes les colonnes et lignes de chaque table source. Sur la page Vérification des abonnements Q de l'assistant, vous pouvez modifier les abonnements Q afin que seul un sous-ensemble des colonnes et des lignes source soit répliqué.</p> <p>Page tables cible</p> <p>Lorsque vous créez plus d'un abonnement Q en une fois, la page tables cible vous permet de réviser le profil objet cible. Modifiez ce profil si nécessaire afin que les tables cible pour les abonnements Q répondent à vos besoins.</p> <p>Le profil objet cible détermine si une table cible existante est utilisée ou si une nouvelle est créée. Le Centre de réplication recherche un objet qui corresponde au système de dénomination défini dans le profil.</p>

Colonnes source d'abonnements Q (réplication unidirectionnelle)

Lorsque vous créez des abonnements Q, les modifications de toutes les colonnes de la table source sont répliquées par défaut vers la table cible ou la procédure stockée. Cependant, si vous créez des abonnements Q destinés à une réplication unidirectionnelle, vous pouvez répliquer un sous-ensemble seulement des colonnes de la table source.

Il est possible que vous souhaitiez répliquer un sous-ensemble des colonnes dans les situations suivantes :

- Vous ne souhaitez pas rendre toutes les colonnes de la table source disponibles pour la table cible ou la procédure stockée.
- La table cible destinée à l'abonnement Q ne prend pas en charge tous les types de données qui sont définis pour la table source.
- En effet, elle existe au préalable et elle contient moins de colonnes que la table source.

La figure 12 montre la réplication d'un sous-ensemble de colonnes.

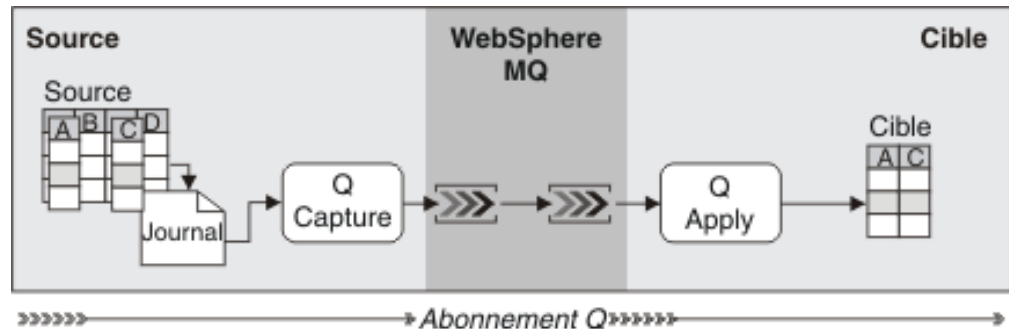


Figure 12. Etablissement de sous-ensembles de colonnes dans un abonnement Q destiné à une réplication unidirectionnelle. Seules les colonnes A et C sont sélectionnées pour réplication. Lorsqu'une ligne est répliquée de la table source vers la table cible, Q Capture réplique uniquement les valeurs des colonnes A et C.

Pour répliquer un sous-ensemble de colonnes, sélectionnez uniquement les colonnes que vous souhaitez répliquer vers la table cible. Si vous créez un seul abonnement Q, les outils d'administration vous offrent différentes possibilités sur la manière de répliquer un sous-ensemble des colonnes de la table source.

Si vous créez plusieurs abonnements Q en même temps dans le Centre de réplication, sélectionnez dans la page Examiner de l'assistant Créer des abonnements Q, l'abonnement Q dans lequel vous souhaitez établir un sous-ensemble de colonnes et éditez les propriétés pour cet abonnement Q.

Particularité des colonnes d'objet LOB : Si vous sélectionnez des colonnes contenant des type de données objet LOB pour un abonnement Q, vérifiez que la table source applique au moins une contrainte d'unicité de base de données (un index à entrées uniques, par exemple, ou une clé primaire). Il n'est pas nécessaire de sélectionner les colonnes qui forment cette propriété d'unicité pour l'abonnement Q.

Fréquence d'envoi d'un message (réplication unidirectionnelle) par le programme Q Capture

Lors de la création d'abonnements Q, vous pouvez spécifier à quel moment le programme Q Capture devra envoyer des messages au programme Q Apply. Il peut le faire soit lorsque des colonnes incluses dans l'abonnement Q sont modifiées, soit chaque fois qu'une colonne dans la table source est modifiée.

Les sections suivantes décrivent les deux types d'événements différents pouvant déclencher l'envoi d'un message par le programme Q Capture :

- «Envoi de message seulement en cas de modification des colonnes faisant partie d'abonnements Q», à la page 73

- «Envoi d'un message lors de toute modification dans la table source»

Si vous répliquez toutes les colonnes de la table source, ces deux options engendrent la même action.

Envoi de message seulement en cas de modification des colonnes faisant partie d'abonnements Q

Par défaut, le programme Q Capture n'envoie un message que si la modification affecte des colonnes sélectionnées pour les abonnements Q.

Par exemple, supposons que votre table source comporte 100 colonnes et que vous en sélectionnez 25 pour une réplication dans un abonnement Q. Si vous avez spécifié de n'envoyer de message que lorsque les colonnes des abonnements Q sont modifiées, à chaque modification d'une colonne parmi les 25 faisant partie de l'abonnement, le programme Q Capture émet un message. Aucun message n'est envoyé en cas de modification d'une des 75 colonnes exclues de l'abonnement.

Recommandation : Ne répliquez que les modifications affectant les colonnes faisant partie d'abonnements Q si des modifications dans les tables source affectent fréquemment des colonnes non abonnées. Utilisez cette option si vous ne désirez pas conserver un historique de toutes les modifications dans les tables source. Cette option minimise le volume de données envoyé à travers les files d'attente.

Envoi d'un message lors de toute modification dans la table source

Vous pouvez définir les abonnements Q de sorte à ce que le programme Q Capture envoie un message en cas de modification quelconque dans la table source. Si vous ne répliquez qu'un sous-ensemble des colonnes de la table source, le programme Q Capture envoie un message même si la modification affecte une colonne ne faisant pas partie de l'abonnement Q.

Par exemple, supposons que votre table source comporte 100 colonnes et que vous en sélectionnez 25 pour une réplication dans un abonnement Q. Si vous avez spécifié d'envoyer un message chaque fois qu'une modification intervient dans la table source, dès lors qu'une modification affecte une colonne quelconque des 100 colonnes de votre table, le programme Q Capture envoie un message.

Recommandation : Utilisez cette option si vous désirez conserver à des fins d'audit un historique de toutes les modifications de la table source.

Conditions de recherche pour filtrer des lignes (réplication unidirectionnelle)

Par défaut, lorsque vous créez des abonnements Q pour une réplication unidirectionnelle, toutes les lignes de la table source sont répliquées vers la table cible ou la procédure stockée. Vous pouvez cependant spécifier une clause WHERE avec une condition de recherche afin d'identifier les lignes que vous souhaitez répliquer.

Lorsque le programme Q Capture identifie un changement dans le journal de récupération DB2 associé à une table source, il évalue ce changement par rapport à la condition de recherche afin de déterminer s'il doit être répliqué vers la table cible ou vers la procédure stockée.

Si vous créez un seul abonnement Q, l'assistant Créer des abonnements Q du Centre de réplication vous permet d'ajouter une clause WHERE pour répliquer un sous-ensemble de lignes de la table source. Si vous créez plusieurs abonnements Q en même temps, sélectionnez alors dans la page Revue de l'assistant Créer des abonnements Q, l'abonnement pour lequel vous souhaitez établir un sous-ensemble de lignes et éditez les propriétés pour cet abonnement pour ajouter la clause WHERE.

Si vous définissez un abonnement Q afin que la table cible soit initialement chargée avec la source, la condition de recherche de l'abonnement est évaluée lorsque la table cible est chargée. Le filtre de lignes étant utilisé pendant le chargement de la table cible, celle-ci contient initialement un sous-ensemble de lignes dans la table source.

Lorsque vous spécifiez une clause WHERE, vous pouvez spécifier si la colonne est évaluée avec des valeurs de l'enregistrement de journal en cours. Si vous souhaitez qu'une colonne soumise à la clause WHERE soit évaluée avec des valeurs de cet enregistrement, mettez deux points devant le nom de la colonne.

Exemple de clause WHERE évaluant une colonne avec des valeurs de l'enregistrement de journal en cours :

```
WHERE :LOCATION = 'EAST' AND :SALES > 100000
```

Dans cet exemple, LOCATION et SALES sont les noms de colonnes de la table source qui sont évaluées. Le programme Q Capture envoie ici uniquement les changements de la table source qui concernent les ventes à l'Est supérieures à 100 000\$. Lorsque vous entrez un nom de colonne, les caractères sont en majuscule sauf si mettez le nom entre guillemets. Par exemple, saisissez "Locat ion" si le nom de la colonne doit respecter la distinction de la casse.

Si Q Capture réplique une colonne soumise à la clause WHERE, il est possible qu'il doive changer le type d'opération devant être envoyé vers la table cible ou la procédure stockée.

Exemple de programme Q Capture devant modifier le type d'opération en raison d'une clause WHERE :

```
WHERE :LOCATION = 'EAST'  
AND :SALES > 100000
```

Supposons que le changement suivant survienne à la table source :

```
INSERT VALUES ( 'EAST', 50000 )  
UPDATE SET SALES = 200000 WHERE LOCATION = 'EAST'
```

La valeur antérieure ne répondant pas à la condition de recherche de la clause, Q Capture envoie l'opération en tant que INSERT au lieu de UPDATE.

De même, si la valeur antérieure répond à la condition de recherche mais pas la valeur après, Q Capture envoie alors les changements UPDATE vers DELETE. Par exemple, si vous avez la même clause WHERE que précédemment :

```
WHERE :LOCATION = 'EAST'  
AND :SALES > 100000
```

Supposons maintenant que le changement suivant survienne à la table source :

```
INSERT VALUES ( 'EAST', 200000 )  
UPDATE SET SALES = 50000 WHERE LOCATION = 'EAST'
```

Le premier changement, l'insertion, est envoyé vers la table cible ou la procédure stockée car il répond à la condition de recherche de la clause WHERE (200000 > 100000 est vrai). Par contre, le deuxième changement, la mise à jour, ne répond pas à cette condition (50000 > 100000 est faux). Q Capture envoie le changement en tant que DELETE afin que la valeur soit supprimée de la table cible ou de la procédure stockée.

Conditions de recherche complexes

La réplication Q vous permet de spécifier des clauses WHERE plus complexes. Cependant, les conditions de recherche complexes peuvent avoir une incidence sur les performances. Vous pouvez par exemple spécifier une clause plus complexe avec une sous-requête qui référence d'autres tables ou enregistrements de la table source ou d'une autre table.

Exemple de clause WHERE avec une sous-requête :

```
WHERE :LOCATION = 'EAST'
AND :SALES > (SELECT SUM(EXPENSE) FROM STORES WHERE STORES.DEPTNO = :DEPTNO)
```

Dans cet exemple, Q Capture envoie uniquement les changements effectués à l'Est ayant généré des bénéfices, où la valeur de la vente est supérieure au montant total des dépenses. La sous-requête référence la table STORES et les colonnes suivantes de la table source : LOCATION, SALES et DEPTNO.

Lorsque vous définissez un abonnement Q avec une sous-requête dans une clause WHERE, vous pouvez rencontrer les problèmes suivants :

- Les performances peuvent être ralenties car, pour chaque changement dans la table source, Q Capture calcule une large sélection de la table STORES pour calculer la valeur SUM(EXPENSE). Ce type de sélection peut également intervenir sur les verrouillages des tables.
- La sous-requête peut produire des résultats inattendus. Par exemple, la sous-requête étant évaluée par rapport aux valeurs de la base de données en cours, l'exemple ci-dessus donne une réponse fautive si la valeur EXPENSE change dans la base de données, tandis que les colonnes de la clause WHERE sont remplacées par les valeurs précédentes de l'enregistrement de journal. Si le nom de la table que la sous-requête référence ne change pas, la condition de recherche fournit alors les bons résultats.

Restrictions des conditions de recherche

- Les conditions de recherche ne peuvent pas contenir de fonctions de colonne à moins que celles-ci n'apparaissent dans une instruction de sous-requête.
- **Clause WHERE non valide avec pour fonctions de colonne :**

```
#-----
# Incorrect : à éviter
#-----
```

```
WHERE :LOCATION = 'EAST' AND SUM(:SALES) > 1000000
```

Le Centre de réplication valide les conditions de recherche lorsque Q Capture les évalue et non pas au moment où le Centre crée l'abonnement Q. Si un abonnement Q contient une condition de recherche non valide, il échouera au moment de la validation de cette condition et sera désactivé.

- Les conditions de recherche ne peuvent contenir une clause ORDER BY ou GROUP BY à moins que cette clause ne soit dans une instruction de sous-requête.

Clause WHERE non valide avec GROUP BY :

```
#-----
# Incorrect : à éviter
#-----
```

```
WHERE :COL1 > 3 GROUP BY COL1, COL2
```

Clause WHERE valide avec GROUP BY :

```
WHERE :COL2 = (SELECT COL2 FROM T2 WHERE COL1=1 GROUP BY COL1, COL2)
```

- Les conditions de recherche ne peuvent pas référencer le nom en cours de la table source à partir de laquelle vous répliquez des changements. N'utilisez pas la notation schema.tablename dans une clause WHERE pour le nom en cours de la table source. Cependant, vous pouvez référencer un autre nom de table dans une sous-requête en utilisant une notation schema.tablename.

Clause WHERE non valide avec pour nom en cours de la table source et nom de colonne :

```
#-----
# Incorrect : à éviter
#-----
```

```
WHERE :ADMINISTRATOR.SALES > 100000
```

Dans la clause WHERE ci-dessus, la table qui est répliquée est ADMINISTRATOR et la colonne a pour nom SALES. Cette clause WHERE non valide doit sélectionner uniquement les valeurs de la colonne SALES de la table administrator, pour laquelle SALES est supérieur à 100000.

Clause WHERE valide avec pour nom de colonne :

```
WHERE :SALES > 100000
```

Ici, SALES est le nom de la colonne.

- Les conditions de recherche ne peuvent référencer les valeurs des colonnes avant changement. Elles ne peuvent les référencer qu'après changement.
- Les conditions de recherche ne peuvent contenir les prédicats EXISTS.
- Elles ne peuvent contenir un prédicat quantifié, qui est un prédicat utilisant SOME, ANY ou ALL.
- Elles ne peuvent pas non plus référencer les valeurs objet LOB.

Variables d'enregistrement de journal pour filtrer des lignes (réplication unidirectionnelle)

Vous pouvez utiliser des variables depuis le journal de récupération de base de données, telles qu'un identificateur utilisateur ou une opération DML, afin de filtrer les lignes à répliquer pour les abonnements Q unidirectionnels. Vous pouvez également combiner ces variables à l'aide d'une clause SQL WHERE afin de créer des filtres de ligne d'une portée plus grande.

Ces variables décrivent la façon dont une modification est survenue dans la table, et elles comprennent des informations sur la procédure de modification, l'utilisateur propriétaire de la transaction et le nom du travail ou du plan z/OS. Vous pouvez utiliser les variables d'enregistrement de journal suivantes :

\$OPERATION	Opération DML. Les valeurs correctes sont I (insert/insérer), U (update/mettre à jour) et D (delete/supprimer).
\$AUTHID	ID utilisateur d'une transaction.

\$AUTHOKEN	z/OS : Clé d'accès (nom de travail) d'une transaction.
\$PLANNAME	z/OS : Nom de plan d'une transaction.

Vous pouvez créer un prédicat utilisant ces variables et sur lequel se base le programme pour déterminer si les transactions doivent être répliquées ou non lorsqu'il lit le journal. Cette fonction permet de créer des filtres qui ne sont pas fondés que les données de la table de base. Par exemple, vous pouvez définir des filtres sur des opérations DML permettant d'alimenter un entrepôt de données, indiquant, par exemple, que seules les opérations d'insertion sont répliquées("\$OPERATION IN 'I'"). Vous pouvez également créer des filtres basés sur des identificateurs au niveau de l'abonnement Q (par exemple, "\$AUTHID = 'HR'"), car la capacité de Q Capture à ignorer certaines transaction ne fonctionne qu'au niveau instance du programme.

Pour indiquer un prédicat, utilisez le programme de ligne de commande ASNCLP avec le mot-clé CHANGE CONDITION de la commande CREATE QSUB ou de la commande ALTER QSUB. Par exemple :

```
CREATE QSUB USING REPLQMAP DALLAS_ASN_TO_TOKYO_ASN
(SUBNAME SALES0002 OPTIONS CHANGE CONDITION
"$AUTHID = 'HR'")
```

Le prédicat est stocké dans la colonne CHANGE_CONDITION de la table IBMQREP_SUBS table.

Combinaison de variables d'enregistrement du journal avec conditions de recherche

Les variables d'enregistrement du journal sont utilisables avec l'autre type de filtre de ligne fourni par Q Capture, à savoir la condition de recherche comportant une clause WHERE de SQL. Contrairement aux valeurs de la condition de recherche, les prédicats basés sur des enregistrements de journal ne possèdent pas le mot-clé WHERE utilisé comme préfixe.

Si vous spécifiez une condition de recherche avec une clause WHERE et un prédicat utilisant des variables d'enregistrement de journal, Q Capture combine les deux prédicats à l'aide de l'opérateur AND. Par exemple, la condition de recherche suivante indique que Q Capture ne réplique que les lignes dont la valeur de la colonne PAYS est FR (suffixe utilisé pour "France" dans les noms de domaine Internet) :

```
WHERE :ETAT = 'FR'
```

Le prédicat suivant comportant des variables d'enregistrement de journal indique Q Capture réplique les transactions avec tous les ID d'autorisation sauf HR :

```
$AUTHID <> 'HR'
```

Lorsque le programme Q Capture détecte la clause WHERE dans la colonne SEARCH_CONDITION de la table IBMQREP_SUBS et le prédicat d'enregistrement de journal dans la colonne CHANGE_CONDITION, il associe les deux en un même prédicat. Q Capture réplique uniquement les lignes dont ETAT est égal à FR et dont l'ID d'autorisation de la transaction est différent de HR :

```
WHERE :ETAT = 'FR' AND $AUTHID <> 'HR'
```

Même si vous indiquez le prédicat combiné entier dans le mot-clé CHANGE CONDITION de la commande CREATE QSUB ou ALTER QSUB, le résultat serait le même (cela est le cas tant que rien n'est indiqué avec le mot-clé SEARCH CONDITION). Cependant, les variables d'enregistrement de journal ne sont utilisables qu'avec CHANGE CONDITION.

Remarque : Bien que la réplication autorise 2048 caractères dans un prédicat CHANGE CONDITION, DB2 risque de ne pas pouvoir traiter tous les prédicats de cette taille. Si l'instruction est trop longue (une fois combinée à SEARCH CONDITION) ou trop complexe, DB2 risque de renvoyer l'erreur SQL0101N ("L'instruction est trop longue ou trop complexe. SQLSTATE=54001").

Remarques relatives aux chargements

Les prédicats avec variables d'enregistrement de journal ne sont pas utilisées lors du chargement de la table cible car les variables s'appuient sur des informations qui n'appartiennent pas à la table source, à partir de laquelle les utilitaires de chargement sélectionnent des données. Les conditions de recherche sont utilisées lors du chargement de la cible car elles indiquent les noms des colonnes de la table source dans lesquelles l'utilitaire de chargement peut effectuer des sélections.

Dans l'exemple ci-dessus, un utilitaire de chargement appelé par le programme Q Apply utiliserait le prédicat suivant de la sélection des données source à placer dans la cible :

```
WHERE :ETAT = 'FR'
```

Si vous souhaitez non seulement utiliser des variables d'enregistrement de journal mais aussi charger la table cible avec un sous-ensemble de données source, veillez à construire correctement vos prédicats.

Remarques relatives à l'utilitaire asntdiff

Si vous employez l'utilitaire asntdiff pour comparer des données dans des tables source et cible, n'oubliez pas que la plupart des opérations d'insertion qu'il recommande de synchroniser sur les tables s'appuient sur le prédicat avec variables d'enregistrement de journal. Evitez d'utiliser asntrep pour réparer la table si vous pensez qu'elle est endommagée car la réparation annulerait l'effet du filtrage à la source.

Comment mapper des colonnes source vers des colonnes cible (réplication unidirectionnelle)

Pour les tables cible existantes, vous pouvez spécifier la manière dont vous voulez mapper les données de colonnes source vers des colonnes cible. Si nécessaire, le centre de réplication crée la table cible (ou un pseudonyme si vous disposez de tables cible non DB2) avec les mêmes colonnes que la table source.

Lorsque vous créez des abonnements Q avec le centre de réplication, vous pouvez choisir parmi les options de mappage suivantes de colonnes source vers des colonnes cible :

Mapper par nom de colonne et type de données

Chaque colonne de la table source est mappée vers la colonne de la table cible (ou le paramètre dans la procédure stockée) ayant les mêmes nom et un type de données.

Mapper par position de colonne et type de données

La première colonne de la table source que vous avez sélectionnée pour être répliquée est mappée vers la première colonne de la table cible ou le paramètre dans la procédure stockée, la deuxième colonne que vous avez sélectionnée est mappée vers la deuxième colonne de la table cible ou le paramètre etc. Les colonnes dans la table cible sont mappées de gauche à droite. Les types de données de chaque couple de colonnes mappées doivent être identiques.

Si des colonnes ne se mappent pas vers des noms de colonnes et des types de données identiques, vous devez alors les mapper manuellement. Cela peut inclure la définition d'une expression SQL pour le mappage de types de données différents.

Remarque : Les règles de mappage créées dans un profil d'objet cible n'affectent pas les abonnements Q existants, mais seulement les abonnements Q ultérieurs.

Vous ne pouvez pas répliquer un nombre de colonnes de la table source supérieur au nombre de colonnes dans la table cible. La table cible peut cependant contenir davantage de colonnes que vous n'en avez sélectionnées dans la table source pour la réplication. Si la table cible contient d'autres colonnes qui ne sont pas mappées vers la table source, ces colonnes ne peuvent pas faire partie de la clé cible et doivent avoir pour valeur Null admis ou bien Non null avec par contre une valeur par défaut.

Tables cible non DB2

Les remarques suivantes concernent le mappage des colonnes source sur les tables dans les bases de données relationnelles non DB2 :

- Si vous choisissez d'avoir une nouvelle table cible créée pour vous, celle-ci sera créée avec des types de données proches du type de données source DB2. Par exemple, si une table source DB2 contient une colonne avec le type de données TIME, une table Oracle sera créée avec une colonne ayant le type de données DATE et le mappage par défaut dans le pseudonyme sera TIMESTAMP. Pour plus de détails sur les mappages utilisés dans le cas des cibles fédérées, consultez la rubrique Mappages de types de données aval par défaut du centre de documentation DB2.
- Si vous choisissez d'effectuer votre réplication vers une table cible existante, les colonnes source sont mappées vers les types de données qui sont définis dans le pseudonyme fédéré DB2 et non vers les types de données de la table cible non DB2.
-

Plusieurs abonnements Q

Si vous créez plusieurs abonnements Q en même temps, vous pouvez utiliser l'assistant Créer des abonnements Q dans le Centre de réplication pour spécifier une règle de mappage des colonnes de la table source vers les colonnes de la table cible ou les paramètres dans la procédure stockée. Lorsque vous créez plusieurs abonnements Q en même temps, le Centre de réplication suppose que vous souhaitez répliquer toutes les colonnes de la table source et il applique par conséquent la règle de mappage des colonnes à toutes les colonnes. Si des colonnes ne se mappent pas selon les règles que vous avez spécifiées, vous devez alors les mapper manuellement.

Procédures stockées

Pour les procédures stockées existantes, vous pouvez spécifier la manière dont vous voulez mapper les données de colonnes source vers des paramètres. La table source ne peut pas répliquer un nombre de colonnes supérieur au nombre de paramètres dans la procédure stockée. Les procédures stockées ne peuvent pas avoir de paramètres supplémentaires non mappées sur des colonnes source.

Utilisation d'expressions dans la réplication Q

Vous pouvez utiliser des expressions pour modifier des données en cours de réplication de la source vers la cible. La réplication Q prend charge les expressions SQL et les expressions XML.

Expressions SQL dans la réplication Q

Lorsque vous créez un abonnement Q pour une réplication unidirectionnelle, vous pouvez utiliser des expressions SQL pour transformer les données entre les tables source et cible, pour mapper plusieurs colonnes source en une colonne cible unique ou pour créer d'autres types de colonnes calculées dans la table cible.

Restrictions

L'utilisation d'expressions SQL n'est pas prise en charge dans les situations suivantes :

- Réplication bidirectionnelle ou réplication entre homologues.
- Colonnes de type LOB
- Cibles de procédure stockée
- Valeurs avant (before) pour les tables CCD
- Fonctions d'agrégation comme SUM et MAX

La réplication Q prend en charge tous les types d'expressions SQL de DB2, y compris les exemples suivants :

Concaténation

Vous pouvez fusionner le contenu de deux colonnes source ou plus en une colonne cible unique. Par exemple, vous pouvez mapper deux colonnes source, COL1 et COL2, sur une colonne cible nommée COLEXP. Au lieu de spécifier une colonne source unique pour l'abonnement Q, vous pouvez spécifier l'expression SQL `CONCAT(:C1,:C2)` et mapper l'expression sur la colonne cible COLEXP.

Sous-chaîne

Vous pouvez utiliser l'expression SQL `substr()` pour appliquer uniquement une portion des données dans une colonne source. Par exemple, vous pouvez utiliser cette fonction pour répliquer des données dans une colonne cible dont la longueur est inférieure à la celle de la colonne source, ou pour extraire des portions d'une adresse et ou de toute autre chaîne de caractères d'une colonne source.

Constante

Vous pouvez remplir les colonnes d'une table cible avec des données provenant des expressions `CONSTANT` et `DERIVED CONSTANT` au lieu de la table source. Par exemple, vous pouvez remplir une colonne avec les caractères IBM, ou la valeur d'un registre DB2 spécial tel que `CURRENT TIMESTAMP`.

Case Vous pouvez utiliser des expressions CASE pour effectuer des transformations de données plus complexes entre les tables source et cible.

La figure 13 montre l'utilisation des expressions dans la réplication Q pour créer des colonnes calculées.

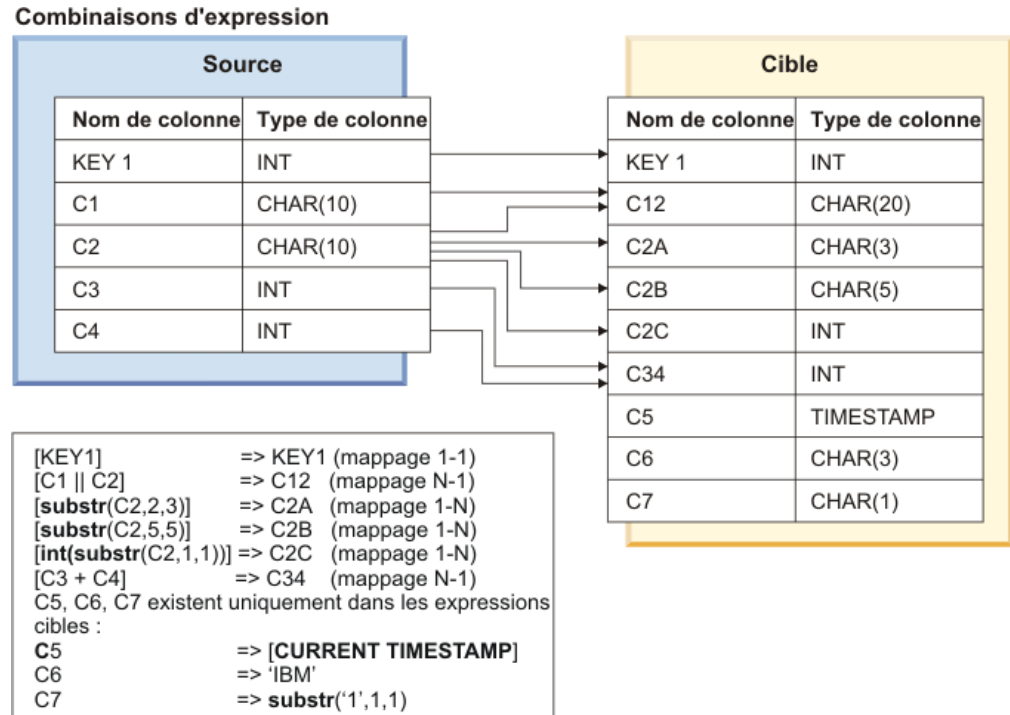


Figure 13. Exemples de prise en charge des expressions SQL pour la réplication Q. Ce diagramme illustre la manière dont les expressions SQL peuvent être utilisées pour créer une table cible contenant différents types de colonnes calculées à partir d'une table source. La légende au dessous de la table source montre les différentes expressions utilisées, comme la concaténation (C1 || C2). Dans la table cible, certaines colonnes de la table source sont mappées sur des colonnes combinées ou des colonnes modifiées. D'autres colonnes de la table cible sont créées à partir d'expressions dérivées de constantes et non des données contenues dans la table source.

Il est possible de mapper un nombre quelconque de colonnes source sur une colonne cible unique, et il est possible de diviser une colonne source unique et de la mapper à un nombre quelconque de colonnes cible.

Une expression peut faire référence à n'importe quelle colonne à partir de la ligne capturée à la source. Vous préfixez le nom de la colonne source de deux points (:) et faites suivre le nom de la colonne d'un espace. Par exemple, :COL1 .

Pour les abonnements Q qui spécifient une phase de chargement pour la table cible, le programme Q Apply utilise DB2 pour évaluer les expressions SQL pendant la phase de chargement et pendant l'application des modifications source depuis la file d'attente auxiliaire. Si vous spécifiez un chargement manuel, vous devez évaluer l'expression au cours du processus de chargement.

Les expressions SQL sont prises en charge par les sources de données, les tables cible fédérées et de modification cohérente des données. Pour des tables CCD, le programme Q Apply évalue les expressions SQL pour les colonnes d'image-avant et d'image-après.

Expressions dans les colonnes de clé

Vous pouvez également spécifier des expressions SQL pour des colonnes qui sont utilisées comme clé de réplication (IS_KEY=Y in the IBMQREP_TRG_COLS table) et pour des colonnes faisant partie d'une contrainte d'unicité au niveau de la table cible mais qui ne sont pas utilisées comme clé de réplication.

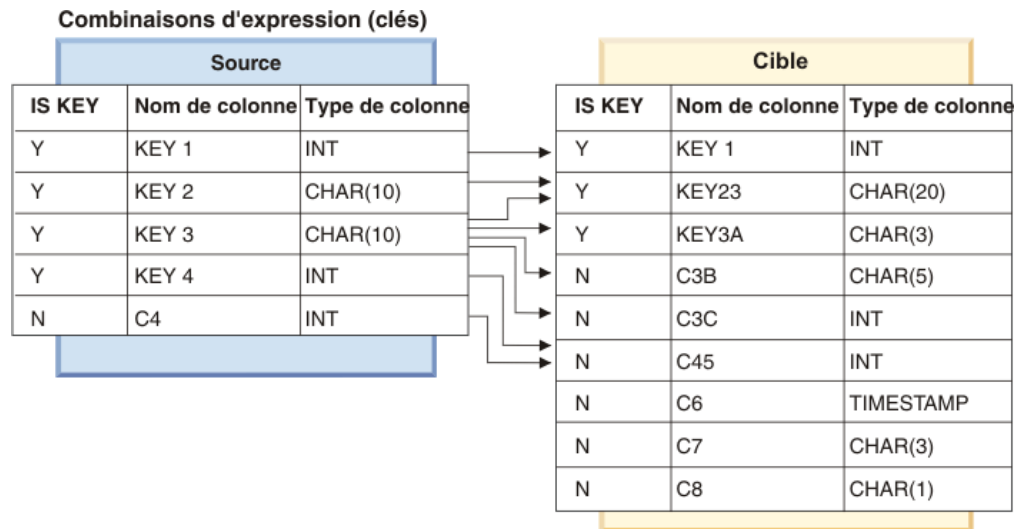
Lorsque vous mappez des colonnes source vers des colonnes cible à l'aide d'expressions SQL, vous pouvez utiliser l'une des combinaisons suivantes :

- Une colonne de clé de source de réplication mappée vers une colonne de clé de cible de réplication
- Un nombre indéfini de colonnes de clé de source de réplication mappées vers une colonne de clé de cible de réplication
- Une colonne de clé de source de réplication divisée et mappée sur un nombre indéfini de colonnes de clé et de non clé de cible de réplication
- Des colonnes de clé et de non clé de réplication au niveau de la source combinées et mappées vers une colonne non clé de cible de réplication

Restrictions :

- Une combinaison de colonnes de clé et de non clé de réplication au niveau de la source mappées vers une colonne de clé de cible de réplication n'est pas prise en charge. Pour résoudre correctement les conflits, le programme Q Apply a besoin que les données soient toujours envoyées pour toute colonne spécifiée comme clé de réplication au niveau de la cible. Si la colonne de clé au niveau de la cible est mappée vers des colonnes de non clé au niveau de la source, il est possible que le programme Q Capture n'envoie pas toujours les données nécessaires, en particulier lorsque la valeur de CHANGED_COLS_ONLY est Y (oui) et que toutes les valeurs antérieures n'ont pas été envoyées.

La figure 14, à la page 83 montre l'utilisation des expressions dans la réplication Q pour créer des colonnes calculées.



[KEY1]	=> KEY1 (1-1 mapping)
[KEY2][KEY3]	=> KEY23 (N-1 mapping)
[substr(KEY3,2,3)]	=> KEY3A (1-N mapping)
[substr(KEY3,5,5)]	=> C3B (1-N mapping)
[init(substr(KEY3,1,1))]	=> C3C (1-N mapping)
[KEY4+C5]	=> C45 (N-1 mapping)
C6, C7, C8 n'existent qu'au niveau de la cible avec des expressions :	
C6	=> [CURRENT_TIMESTAMP]
C7	=> 'IBM'
C8	=> substr('1',1,1)

Figure 14. Exemples de prise en charge des expressions SQL dans les colonnes clés pour la réplication Q. Ce diagramme illustre la manière dont les expressions SQL peuvent être utilisées pour créer une table cible contenant différents types de colonnes calculées à partir d'une table source. La légende au-dessous de la table source montre les différentes expressions utilisées, comme la sous-chaîne [substr(KEY3,2,3)]. Dans la table cible, certaines colonnes de la table source sont mappées sur des colonnes combinées ou des colonnes modifiées. D'autres colonnes de la table cible sont créées à partir d'expressions dérivées de constantes et non des données contenues dans la table source.

L'index ou la clé que vous spécifiez pour un abonnement Q ne doit pas obligatoirement correspondre entre la source et la cible. Lorsque vous choisissez des colonnes comme clés pour la réplication, vous devez choisir toutes les colonnes qui correspondent à l'une des contraintes d'unicité au niveau de la cible. Cet exigence permet au programme Q Apply de détecter et de résoudre correctement les conflits. Si aucune contrainte d'unicité n'existe au niveau de la cible, vous devez alors choisir toutes les colonnes de la cible faisant partie de la clé de réplication (en excluant certains types de colonne, comme LOB et LONG).

Prise en charge de l'expression dans les outils d'administration de réplication

Si vous utilisez le programme de ligne de commande ASNCPLP pour créer un abonnement Q, vous utilisez le mot-clé EXPRESSION, qui fait partie de l'option TRGCOLS. Vous spécifiez l'expression SQL et le nom de la colonne cible sur laquelle l'expression se mappe.

Par exemple, les jeux de commandes suivants définissent l'environnement et créent un abonnement Q avec une nouvelle table cible qui comporte toutes les colonnes de la table source et spécifie une expression qui concatène les colonnes COL1 et COL2 et les mappe sur la colonne de la table cible CEXP.

```

ASNCLP SESSION SET TO Q REPLICATION;
SET RUN SCRIPT NOW STOP ON SQL ERROR ON;
SET SERVER CAPTURE TO DB SAMPLE;
SET SERVER TARGET TO DB TARGET;
SET CAPTURE SCHEMA SOURCE ASNCAP1;
SET APPLY SCHEMA ASNAPP1;
CREATE QSUB USING REPLQMAP SAMPLE ASNCAP1 TO TARGET ASNAPP1
(SUBNAME TESTEXP DATA.EMPLOYEE TARGET NAME DATA.TGTEMPLOYEE
TRGCOLS ALL EXPRESSION ("CONCAT(:COL1,:COL2)" TARGET CEXP));

```

Vous pouvez également spécifier des expressions SQL lorsque vous modifiez un abonnement Q existant à l'aide de la commande ALTER QSUB.

Dans le Centre de réplication, vous pouvez créer des colonnes calculées en utilisant la fenêtre Expression. Vous accédez à la fenêtre Expression à partir de la fenêtre Mappage de colonne, qui est ouverte à partir de la page Lignes et colonnes de l'assistant Créer des abonnements Q. Cliquez sur **Valider** pour vérifier la syntaxe de l'expression.

Expressions SQL au niveau des octets

Si les tables source et cible présentent des pages de code différentes, les expressions SQL qui évaluent les données au niveau des octets (par exemple, une fonction HEX) pourraient produire des résultats différents dans la table cible selon que les données sont répliquées lors d'un chargement automatique ou lors d'opérations régulières du programme Q Apply :

Lors d'un chargement automatique

DB2 évalue l'expression SQL sur les données source et convertit ensuite les résultats sur la page de codes cible.

Lors de la réplication d'une ligne

Le programme Q Apply convertit les données source vers la page de codes cible et DB2 évalue ensuite l'expression SQL pour la ligne.

Une alternative à l'utilisation de la fonction HEX consiste à définir la colonne cible comme FOR BIT DATA. Le programme Q Apply évite ensuite la conversion de page de codes et les données source sont sauvegardées dans la table cible dans une représentation hexadécimale.

Expressions XML dans la réplication Q

Lorsque vous créez un abonnement Q pour une réplication unidirectionnelle, vous pouvez utiliser des expressions XML pour transformer les données XML entre les tables source et cible.

La réplication Q prend en charge le type de données XML. La prise en charge des expressions dans les colonnes XML permet d'effectuer des transformations importantes avec des données XML. Par exemple, vous pouvez modifier la forme d'un objet XML, ne répliquer qu'une sous-partie de celui-ci et le rendre conforme à un autre schéma sur la cible. Les expressions sur des types de données XML peuvent être requises dans certaines situations, lorsque l'application sur la cible a besoin de données à un autre format. C'est le cas, par exemple, des migrations d'applications, de l'alimentation d'un entrepôt de données et de l'intégration d'applications.

Vous pouvez utiliser le Centre de réplication ou le programme de ligne de commande ASNCLP pour définir des expressions XML lorsque vous créez un abonnement Q, de la même façon que vous définissez des expressions SQL.

- «Fonctions SQL/XML DB2 prises en charge »
- «Restrictions»
- «Exemples d'expressions XML», à la page 86

Fonctions SQL/XML DB2 prises en charge

Le programme Q Apply intègre des expressions XML dans les instructions SQL qu'il utilise pour mettre à jour les tables cible. Q Apply n'analyse pas les expressions pour déterminer si elles sont syntaxiquement et sémantiquement correctes. Si l'utilisation est incorrecte, Q Apply rapporte l'avertissement ou l'erreur SQL DB2 dans la table IBMQREP_EXCEPTIONS lorsque le programme tente d'exécuter l'instruction et suit l'action définie en cas d'erreur pour l'abonnement Q.

Recommandation : Testez l'instruction SQL pour valider sa syntaxe avant de l'ajouter à un abonnement Q.

Le tableau 8 fournit des exemples d'expressions XML prises en charge et répertorie les expressions non prises en charge.

Tableau 8. Expressions XML prises en charge et non prises en charge

Exemples d'expressions XML prises en charge	Expressions XML non prises en charge
<ul style="list-style-type: none"> • Fonction scalaire XMLATTRIBUTES • Fonction scalaire XMLCOMMENT • Spécification XMLCAST • Fonction scalaire XMLCONCAT • Fonction scalaire XMLDOCUMENT • Fonction scalaire XMLELEMENT • Fonction scalaire XMLFOREST • Déclaration XMLNAMESPACES • Fonction scalaire XMLPARSE • Fonction scalaire XMLPI • Fonction scalaire XMLQUERY • Fonction scalaire XMLROW • Fonction scalaire XMLSERIALIZE • Fonction scalaire XMLTEXT • Fonction scalaire XMLVALIDATE 	<p>Certaines de ces fonctions, la fonction d'agrégation XMLAGG par exemple, ne sont pas prises en charge car elles utilisent des valeurs XML d'autres lignes et Q Apply gère les données une ligne à la fois. D'autres fonctions peuvent être utilisées dans des requêtes mais pas en tant que composant d'une expression.</p> <ul style="list-style-type: none"> • Fonction scalaire PARAMETER : vous pouvez utiliser cette fonction dans la fonction db2-fn:sqlquery. Une requête ne fait pas partie d'une expression. • Fonction agrégation XMLAGG • Prédicat XMLEXISTS : peut être utilisé dans la clause WHERE d'une requête • Fonction d'agrégation XMLGROUP • Fonction d'agrégation XMLTABLE • Fonction scalaire XMLXSROBJECTID sans valeur de sortie XML • Fonction scalaire XMLTRANSFORM : non prise en charge car des sorties autres qu'une valeur XML sont possibles.

Restrictions

Les restrictions suivantes s'appliquent à la prise en charge des expressions XML par la réplication Q :

- Vous ne pouvez pas répliquer des expressions XML à la fois sur des colonnes XML et des colonnes clé.
- Vous ne pouvez pas mapper des colonnes d'objets LOB (CLOB, BLOB, DBCLOB) vers des colonnes XML.

- Vous ne pouvez pas mapper des colonnes XML vers des colonnes qui ne sont pas des colonnes d'objets LOB ou vers des colonnes non XML, comme des colonnes CHAR et VARCHAR.
- Plusieurs fonctions ne sont pas prises en charge lorsqu'elles sont utilisées sur plusieurs plateformes DB2 :
 - XMLQUERY n'est pas prise en charge lorsque la source est DB2 pour z/OS et que la cible est DB2 pour Linux, UNIX et Windows.
 - XMLQUERY sous DB2 pour Linux, UNIX et Windows prend en charge les expressions du langage XQuery tandis que XMLQUERY sous DB2 pour z/OS prend en charge les expressions du langage XPath.
 - XMLVALIDATE sous DB2 pour Linux, UNIX et Windows a une syntaxe différente de la fonction correspondante DSN_XMLVALIDATE sous DB2 pour z/OS. De plus, le schéma qui décrit le document transformé doit être enregistré à la fois sur la source et sur la cible. L'opération LOAD extrait les données transformées de la source avant d'appliquer les données transformées à la cible. Si le schéma n'est pas enregistré pour être utilisé sur la source, l'opération LOAD est arrêtée et l'abonnement Q est désactivé.

Exemples d'expressions XML

Les exemples suivants illustrent les différentes façons dont les expressions XML peuvent être utilisées entre la source et la cible :

Concaténation de documents XML et ajout d'un nouveau noeud racine

Vous pouvez utiliser des expressions pour fusionner les données de deux colonnes XML dans un document XML encadré par un nouveau noeud racine. L'exemple suivant associe les données des colonnes xmlcol1 et xmlcol2 de la source à l'aide de la fonction XMLCONCAT et encadre les données avec un nouvel élément racine à l'aide de la fonction XMLELEMENT. Dans cas, vous mappez les colonnes xmlcol1 et xmlcol2 à une colonne cible unique et vous définissez l'expression.

Données source	xmlcol1: <book><author>Foo</author><title>bar</title></book> xmlcol2: <book><author>Bar</author><title>Foo</title></book>
Expression	XMLELEMENT(NAME "bookstore", XMLCONCAT(:xmlcol1, :xmlcol2))
Données obtenues sur la cible	<bookstore> <book><author>Foo</author><title>bar</title></book> <book><author>Bar</author><title>Foo</title></book> </bookstore>

Transformation ou définition de sous-ensembles de données

Vous pouvez utiliser XQUERY pour modifier les documents XML répliqués. Par exemple, vous pouvez supprimer un élément du document XML, ou y ajouter de nouveaux éléments. L'exemple suivant supprime l'élément <title> et son contenu du document XML source :

Données source	<book><author>Foo</author><title>bar</title></book>
Expression	XMLQUERY(copy \$oldval := \$d modify do delete \$oldval/book/title return \$oldval' PASSING :xmlcol AS "d")

Données obtenues sur la cible	<book><author>Foo</author></book>
--------------------------------------	-----------------------------------

Validation des documents XML finaux

Il peut être nécessaire de valider un document XML modifié par rapport à un schéma donné situé dans la base de données cible. A l'aide de la fonction XMLVALIDATE dans une expression, vous pouvez effectuer la validation lors de la réplication des données. L'expression doit contenir XMLVALIDATE autour d'une autre expression indiquant l'emplacement du schéma comme paramètre, comme illustré dans l'exemple suivant :

Données source	<book><author>Foo</author><title>bar</title></book>
Expression	XMLVALIDATE(XMLQUERY('transform copy \$oldval := \$d modify do delete \$oldval/book/title return \$oldval' PASSING :xmlcol AS "d") ACCORDING TO XMLSCHEMA ID modifiedBooks.xsd)
Données obtenues sur la cible	<book><author>Foo</author></book>

Mappage de colonnes CHAR et VARCHAR vers des colonnes XML

Vous pouvez migrer des données source CHAR et VARCHAR contenant des documents XML dans le format XML natif DB2. La fonction XMLPARSE analyse les données de type caractères et la réplication stocke les données dans une colonne XML sur la cible, comme illustré dans l'exemple suivant :

Données source	<book><author>Foo</author><title>bar</title></book>
Expression	XMLPARSE(DOCUMENT :xmlcol)
Données obtenues sur la cible	<book><author>Foo</author><title>bar</title></book>

Index ou colonnes de clé des cibles (réplication unidirectionnelle)

Le programme Q Apply utilise une clé primaire, une contrainte d'unicité ou un index à entrées uniques dans la table cible pour appliquer l'unicité de chaque ligne lorsqu'il les applique aux tables cible ou aux paramètres dans les procédures stockées.

Si vous créez un seul abonnement Q, l'assistant Créer des abonnements Q du Centre de réplication vous permet alors de sélectionner les colonnes qui identifient de manière unique les lignes dans la cible. Si vous créez plusieurs abonnements Q en même temps, vous pouvez utiliser la page Revue de l'assistant Créer des abonnements Q pour personnaliser les colonnes qui sont utilisées de manière unique dans la cible.

Si le Centre de réplication trouve une unicité dans la table source, il vous recommande un index ou une table cible. Vous pouvez accepter la recommandation ou indiquer les colonnes désirées comme clés de réplication pour la cible.

Restriction : Les colonnes d'objets LOB (Large object) et les colonnes LONG ne peuvent pas être utilisées comme clés (IS_KEY dans les tables de contrôle IBMQREP_SRC_COLS IBMQREP_TRG_COLS).

La recommandation du Centre de réplication dépend si la table cible existe déjà. Les sections suivantes expliquent la logique du Centre de réplication dans la recommandation d'une clé ou d'un index cible :

- «Clé ou index cible destiné à de nouvelles tables cible»
- «Clé ou index cible destiné à des tables cible existantes dans des abonnements Q»

Clé ou index cible destiné à de nouvelles tables cible

Lorsque le Centre de réplication recommande une clé primaire, une contrainte d'unicité ou un index à entrées uniques pour de nouvelles tables cible, il vérifie la table source pour l'une des définitions suivantes et dans l'ordre suivant :

1. Une clé primaire
2. Une contrainte d'unicité
3. Un index à entrées uniques

Si le Centre de réplication trouve l'une de ces définitions pour la table source et si ces colonnes source sont sélectionnées pour réplication, le Centre de réplication utilise alors la clé primaire de la table source (ou la contrainte d'unicité ou l'index à entrées uniques) comme clé de la table cible. Lorsque le Centre génère le langage SQL pour construire les nouvelles tables cible, il construit ces dernières avec la clé ou l'index que vous avez spécifié. Vous pouvez utiliser le nom et le schéma de l'index par défaut ou bien modifier les valeurs par défaut pour qu'elles correspondent à vos conventions de dénomination.

Si le Centre de réplication ne trouve pas de clé primaire, de contrainte d'unicité ou d'index à entrées uniques dans la table source, il crée automatiquement un index à entrées uniques pour la nouvelle table cible, basé sur toutes les colonnes source valides et abonnées.

Clé ou index cible destiné à des tables cible existantes dans des abonnements Q

Lorsque que le Centre de réplication recommande une clé primaire, une contrainte d'unicité ou un index à entrées uniques pour des tables cible existantes, il vérifie en premier lieu si une clé primaire, une contrainte d'unicité ou un index à entrées uniques existe déjà dans ces tables. S'il trouve une unicité dans la table cible, il s'assure que les colonnes concernées font partie de celles que vous avez choisies de répliquer à partir de la table source. Il peut également vérifier si la table source utilise ces colonnes pour appliquer cette unicité à cette table. Si les tables source et cible ont au moins une correspondance exacte pour les colonnes de clé ou d'index, le Centre recommande alors d'utiliser ces colonnes pour établir une unicité de lignes cible.

Si aucune unicité n'existe dans la table cible, il vérifie la table source pour l'une des définitions suivantes et dans l'ordre suivant :

1. Une clé primaire
2. Une contrainte d'unicité
3. Un index à entrées uniques

S'il trouve l'une de ces définitions pour la table source et si ces colonnes source sont sélectionnées pour réplification, il recommande alors la clé primaire, la contrainte d'unicité ou l'index à entrées uniques de la table source.

Options de traitement de conditions imprévues dans la table cible (réplication unidirectionnelle)

Le programme Q Apply met à jour les tables cible avec les changements qui sont intervenus dans la table source. Si d'autres applications réalisent également des changements dans la table cible, il est possible que Q Apply rencontre des lignes dans cette table différentes de celles attendues.

Par exemple, Q Apply peut essayer de mettre à jour dans la table cible une ligne déjà supprimée par une autre application. L'option que vous choisissez dépend du niveau de granularité auquel vous souhaitez isoler et résoudre l'incident.

Il est fort probable que, dans beaucoup de scénarios, vous souhaitiez que Q Apply force les changements ou ignore des conditions inattendues dans les données cibles. Pour certains scénarios cependant, il est possible que vous ne prévoyiez jamais de problèmes dans les données cibles et par conséquent que vous choisissiez une action différente, selon le niveau auquel vous pensez avoir besoin d'identifier les incidents.

Vous pouvez spécifier que Q Apply exécute l'une des actions suivantes lorsqu'il rencontre des conditions inattendues dans des données cibles :

- «Forcer les changements», à la page 90
- «Ignorer la condition inattendue», à la page 90
- «Désactiver l'abonnement Q correspondant», à la page 90
- «Interrompre la lecture du programme Q Apply dans la file d'attente de réception correspondante», à la page 91
- «Arrêter le programme Q Apply», à la page 92

Tables cible de modification cohérente des données (CCD) : si la table CCD est condensée et complète (CONDENSED=Y et COMPLETE=Y), vous pouvez choisir de forcer le changement ou d'ignorer la condition inattendue. Pour tous les autres types de table CCD, forcer le changement est le seul choix valide.

Procédures stockées : si Q Apply fait passer les données à une procédure stockée, l'action que vous devez sélectionner dépend quant à elle du comportement de cette procédure. Dans la plupart des scénarios où Q Apply transmet les données à une procédure stockée, vous ne souhaitez pas forcer les changements en transformant l'opération sur les lignes. Dans la plupart des scénarios comprenant des procédures stockées qui traitent des données, il est possible que vous souhaitiez spécifier que Q Apply ignore les conditions inattendues dans les données cible. Néanmoins, si votre procédure stockée reconstitue les instructions SQL pour la ligne concernée et ne transforme que certaines données (uniquement des dollars américains en euros, par exemple), il est alors possible que vous envisagiez de forcer les changements.

Quelle que soit l'option choisie pour traiter ces conditions inattendues, chaque fois que Q Apply rencontre une erreur lors du traitement d'une ligne, il la consigne

dans la table IBMQREP_APPLYTRACE et son fichier journal diagnostic. De même, une copie de la ligne en erreur est insérée dans la table IBMQREP_EXCEPTIONS.

Forcer les changements

Lorsque Q Apply rencontre une condition inattendue dans les données cible, il force les changements de la table source dans la table cible ou les paramètres pour la procédure stockée. Q Apply modifie l'opération (une mise à jour au lieu d'une insertion, par exemple) afin qu'elle puisse être appliquée à la table cible ou passée aux paramètres de la procédure stockée. Il essaie ensuite d'appliquer à nouveau le changement.

Recommandation : Il peut être utile d'indiquer au programme Q Apply de forcer les modifications si les données qu'il reçoit de la table source correspondent toujours à ce que vous voulez dans la table cible.

Ignorer la condition inattendue

Lorsque Q Apply rencontre une condition inattendue dans les données cible, il l'ignore, il n'applique pas la ligne concernée, il enregistre l'erreur ainsi que toutes les lignes qu'il n'a pas appliquées et il termine puis valide le reste de la transaction. Quelles que soient les données cibles, elles ne sont pas écrasées. Néanmoins, si vous choisissez que Q Apply ignore les conditions inattendues, certaines données ne sont pas appliquées dans la table cible. Les lignes n'ont appliquées sont alors toutes enregistrées dans la table IBMQREP_EXCEPTIONS.

Recommandation : Il est possible que vous souhaitiez que Q Apply ignore les conditions inattendues dans les données cible dans les situations suivantes :

- La convergence des données des tables source et cible n'est pas importante dans votre scénario.
- Tous les états SQL sont attendus et peuvent être tolérés.

Désactiver l'abonnement Q correspondant

Lorsque Q Apply rencontre une condition inattendue dans les données cible, il désactive (ou arrête) uniquement l'abonnement Q où cette condition est rencontrée et il continue d'appliquer les changements aux autres abonnements Q. Il consigne l'erreur et toutes les lignes qu'il n'a pas appliquées puis complète et valide le reste de la transaction. Le programme Q Capture arrête de capturer les modifications qui se produisent sur la table source pour un abonnement Q désactivé. Cette option vous permet d'obtenir le meilleur niveau de granularité pour les problèmes d'identification des incidents dans une table particulière. Vous pouvez vérifier le problème survenu pour l'abonnement Q, puis activer ce dernier une fois le problème résolu. N'oubliez pas que, lorsqu'un abonnement Q est désactivé, il faut de nouveau charger sa cible au moment où vous le réactivez.

Recommandations :

- Il est possible que vous souhaitiez désactiver l'abonnement Q correspondant lorsque des conditions imprévues sont rencontrées dans les situations suivantes :
 - Vous avez plusieurs abonnements Q définis sur des tables non associées qui sont répliquées à l'aide du même mappe de files de réplication.
 - Peu de changements se produisent dans la table (en particulier si elle est parent et parce que les changements se produisent rarement dans les colonnes de clés parentes).

- Il est possible que vous souhaitiez désactiver l'abonnement Q correspondant lorsque des conditions imprévues sont rencontrées dans les situations suivantes :
 - Il y a une intégrité référentielle entre la table de l'abonnement Q et les autres tables. Par conséquent, la désactivation de cet abonnement peut avoir un impact sur ces tables. Si la table est une table enfant, il est possible que les erreurs d'intégrité référentielle fassent échouer dans la table enfant les suppressions ou les mises à jour effectuées dans la table parent (si DELETE RESTRICT est activée). Il est possible que tous les abonnements Q aient finalement des erreurs liées à cette intégrité référentielle qui auront pour conséquence la désactivation des abonnements.
 - Au moment de charger à nouveau l'abonnement Q lorsque ce dernier est réactivé, vous ne souhaitez pas charger à nouveau la cible.
 - Désactiver un abonnement Q a un résultat trop lourd dans votre scénario si une erreur se produit.

Interrompre la lecture du programme Q Apply dans la file d'attente de réception correspondante

Lorsqu'il rencontre une condition imprévue, Q Apply arrête d'appliquer les changements pour tous les abonnements Q sur la file d'attente de réception et pas uniquement pour l'abonnement contenant l'erreur en question. Il consigne l'erreur ainsi que toutes les lignes qu'il n'a pas appliquées mais il ne complète pas, ni ne valide, le reste de la transaction. Toutes les transactions touchées par cette condition imprévue sont annulées.

Q Apply continue de lire à partir de n'importe quelles autres files d'attente de réception mais il n'essaie pas de le faire à partir de la file d'attente désactivée. Le programme Q Capture continue d'envoyer des données à la file d'attente désactivée, vous devez donc corriger le problème et redémarrer la file d'attente de réception avant que celle-ci ne soit pleine. Si le programme Q Capture ne peut plus enregistrer de données sur la file d'attente d'envoi, le programme Q Capture désactive tous les abonnements Q qui utilisent la file d'attente d'envoi ou il s'arrête, en fonction de l'option d'erreur que vous avez spécifiée pour la mappe de file d'attente de réplication.

Recommandations :

- Il est possible que vous souhaitiez que Q Apply n'applique plus de transactions à partir de la file d'attente de réception lorsque des conditions inattendues sont rencontrées dans les situations suivantes :
 - Les conditions imprévues dans les données cible ne sont pas tolérées dans votre scénario et vous n'en prévoyez pas beaucoup.
 - Vous avez plusieurs abonnements Q définis sur des tables non associées et partageant le même mappe de files de réplication. Par conséquent, vous souhaitez que tous les abonnements s'arrêtent lorsque l'un d'entre eux comprend une condition inattendue dans les données cible. Vous pouvez vérifier le problème survenu pour cet abonnement puis, après avoir résolu ce problème, vous pouvez redémarrer la file d'attente de réception. Cette option est adaptée si peu de changements sont répliqués vers la table cible car elle permet de préserver les transactions et de maintenir une intégrité référentielle entre les tables associées.

- Il est possible que vous ne souhaitiez pas que Q Apply arrête d'appliquer les transactions à partir de la file d'attente de réception si le résultat obtenu est trop lourd pour votre scénario lorsqu'une condition inattendue est rencontrée dans les données cible.

Arrêter le programme Q Apply

Lorsque le programme Q Apply rencontre une condition inattendue dans les données cible, il s'arrête mais la file d'attente de réception continue de recevoir les données source du programme Q Capture. Il consigne l'erreur ainsi que toutes les lignes qu'il n'a pas appliquées mais il ne complète pas, ni ne valide, le reste de la transaction. Toutes les transactions touchées par cette condition inattendue sont annulées. Si Q Apply lit à partir d'une seule file d'attente de réception, le résultat de cette option est identique à celui obtenu lorsque Q Apply n'applique plus les changements pour tous les abonnements sur la file d'attente de réception.

L'arrêt du programme est la réponse aux problèmes d'erreurs la plus grave que vous puissiez définir pour le programme Q Apply. Le programme Q Capture continue d'envoyer des données vers la file d'attente de réception, vous devez donc corriger l'incident et redémarrer la file d'attente de réception avant qu'elle ne soit pleine. Si le programme Q Capture ne peut plus enregistrer de données sur la file d'attente d'envoi, le programme Q Capture désactive tous les abonnements Q qui utilisent la file d'attente d'envoi ou il s'arrête, en fonction de l'option d'erreur que vous avez spécifiée pour la mappe de file d'attente de réplication.

Recommandations :

- Il est possible que vous souhaitiez désactiver l'abonnement Q correspondant lorsque des conditions imprévues sont rencontrées dans les situations suivantes :
 - Vous avez plusieurs abonnements Q définis sur des tables non associées et les données sont transportées sur plusieurs mappes de files de réplication. Par conséquent, vous souhaitez que tous les abonnements s'arrêtent lorsque l'un d'entre eux comprend une condition inattendue dans les données cible. Vous pouvez vérifier le problème survenu pour cet abonnement puis, après avoir résolu ce problème, vous pouvez redémarrer la file d'attente de réception. Cette option est adaptée si peu de changements sont répliqués vers la table cible car elle permet de préserver les transactions et de maintenir une intégrité référentielle entre les tables associées.
 - Vous souhaitez pouvoir surveiller facilement votre configuration. Cette option pour pouvoir arrêter Q Apply est identique à celle où Q Apply arrête d'appliquer les transactions à partir de la file d'attente de réception. Cependant, il est possible que votre environnement soit plus facile à surveiller si Q Apply s'arrête que s'il arrête simplement de lire à partir d'une file d'attente de réception précise. Vous pouvez par exemple choisir une telle option lorsque vous procédez à des tests.
- Il est possible que vous ne souhaitiez pas que Q Apply s'arrête si le résultat obtenu est trop lourd pour votre scénario lorsqu'une condition inattendue est rencontrée dans les données cible.

Options relatives aux erreurs pour la réplication Q

Dans la réplication Q, vous pouvez définir les actions exécutées par le programme Q Apply lorsqu'il rencontre des erreurs dans votre environnement, telles que les erreurs SQL. L'option que vous choisissez dépend du niveau de granularité auquel

vous souhaitez isoler et résoudre l'incident. Les options applicables à la réplication unidirectionnelle et à la réplication multidirectionnelle sont les mêmes.

Vous pouvez choisir, parmi les actions suivantes, celle que vous souhaitez activer lorsque le programme Q Apply rencontre des erreurs :

- «Désactiver l'abonnement Q correspondant»
- «Interrompre la lecture du programme Q Apply dans la file d'attente de réception correspondante», à la page 94
- «Arrêter le programme Q Apply», à la page 94

Quelle que soit l'option choisie pour traiter les erreurs, chaque fois que Q Apply rencontre une erreur, il la consigne dans la table `IBMQREP_APPLYTRACE` et ses fichiers journaux diagnostic. Une copie de toutes les lignes qui n'ont pas été appliquées et les détails des erreurs sont également insérés dans la table `IBMQREP_EXCEPTIONS`.

Désactiver l'abonnement Q correspondant

Lorsque Q Apply rencontre une erreur, il désactive (ou arrête) uniquement l'abonnement Q où l'erreur s'est produite et continue d'appliquer les changements aux autres abonnements Q. Il consigne l'erreur et toutes les lignes qu'il n'a pas appliquées puis complète et valide le reste de la transaction. Le programme Q Capture arrête de capturer les modifications qui se produisent sur une table source pour l'abonnement Q désactivé. Cette option vous permet d'obtenir le meilleur niveau de granularité pour les problèmes d'identification des incidents dans une table particulière. Vous pouvez vérifier le problème survenu pour l'abonnement Q, puis activer ce dernier une fois le problème résolu. N'oubliez pas que, lorsqu'un abonnement Q est désactivé, il faut de nouveau charger sa cible au moment où vous le réactivez.

Recommandations :

- Il est possible que vous souhaitiez désactiver l'abonnement Q correspondant lorsqu'une erreur se produit dans les situations suivantes :
 - Vous avez deux abonnements Q définis sur des tables non associées qui sont répliquées à l'aide de la même mappe de files de réplication.
 - Peu de changements se produisent dans la table (en particulier si elle est parent et parce que les changements se produisent rarement dans les colonnes de clés parentes).
- Il est possible que vous ne souhaitiez pas désactiver l'abonnement Q correspondant lorsqu'une erreur se produit dans les situations suivantes :
 - Il y a une intégrité référentielle entre la table de l'abonnement Q et les autres tables. Par conséquent, la désactivation de cet abonnement peut avoir un impact sur ces tables. Si la table est une table enfant, il est possible que les erreurs d'intégrité référentielle fassent échouer dans la table enfant les suppressions ou les mises à jour effectuées dans la table parent (si `DELETE RESTRICT` est activée). Il est possible que tous les abonnements Q aient finalement des erreurs liées à cette intégrité référentielle qui auront pour conséquence la désactivation des abonnements.
 - Au moment de charger à nouveau l'abonnement Q lorsque ce dernier est réactivé, vous ne souhaitez pas charger à nouveau la cible.
 - Désactiver un abonnement Q a un résultat trop lourd dans votre scénario si une erreur se produit.

Interrompre la lecture du programme Q Apply dans la file d'attente de réception correspondante

Lorsqu'il rencontre une erreur, Q Apply arrête d'appliquer les changements pour tous les abonnements Q sur la file d'attente de réception et pas uniquement pour l'abonnement contenant l'erreur en question. Il consigne l'erreur ainsi que toutes les lignes qu'il n'a pas appliquées mais il ne complète pas, ni ne valide, le reste de la transaction. Toutes les transactions touchées par cette erreur sont annulées. Si Q Apply procède à la lecture à partir d'autres files d'attente de réception, il continue alors de traiter les données à partir de ces files d'attente. S'il ne procède pas à la lecture à partir d'autres files d'attente de réception, il s'arrête. Le programme Q Capture continue d'envoyer des données vers la file d'attente de réception, vous devez donc corriger l'incident et redémarrer la file d'attente de réception avant qu'elle ne soit pleine. Si le programme Q Capture ne peut plus enregistrer de données sur la file d'attente d'envoi, le programme Q Capture désactive tous les abonnements Q qui utilisent la file d'attente d'envoi ou il s'arrête, en fonction de l'option d'erreur que vous avez spécifiée pour la mappe de file d'attente de réplication.

Recommandations :

- Il est possible que vous souhaitiez que Q Apply n'applique plus de transactions à partir de la file d'attente de réception lorsqu'une erreur se produit dans les situations suivantes :
 - Les erreurs ne sont pas tolérées dans votre scénario et vous espérez qu'il ne s'en produira pas souvent.
 - Vous avez plusieurs abonnements Q définis sur des tables non associées et partageant le même mappe de files de réplication. Par conséquent, vous souhaitez que tous les abonnements s'arrêtent lorsque l'un d'entre eux contient une erreur. Vous pouvez vérifier le problème survenu pour cet abonnement puis, après avoir résolu ce problème, redémarrer la file d'attente de réception. Cette option est adaptée si peu de changements sont répliqués vers la table cible car elle permet de préserver les transactions et de maintenir une intégrité référentielle entre les tables associées.
- Il est possible que vous ne souhaitiez pas que Q Apply arrête d'appliquer les transactions à partir de la file d'attente de réception si le résultat obtenu est trop lourd pour votre scénario lorsqu'une erreur se produit.

Arrêter le programme Q Apply

Lorsque le programme Q Apply rencontre une erreur, il s'arrête, mais la file d'attente de réception continue de recevoir des données source du programme Q Capture. Il consigne l'erreur ainsi que toutes les lignes qu'il n'a pas appliquées mais il ne complète pas, ni ne valide, le reste de la transaction. Toutes les transactions touchées par cette erreur sont annulées. Si Q Apply lit à partir d'une seule file d'attente de réception, le résultat de cette option est identique à celui obtenu lorsque Q Apply n'applique plus les changements pour tous les abonnements sur la file d'attente de réception.

L'arrêt du programme est la réponse aux problèmes d'erreurs la plus grave que vous puissiez définir pour le programme Q Apply. Le programme Q Capture de ces abonnements Q continue d'envoyer des données vers la file d'attente de réception, vous devez donc corriger le problème et redémarrer la file d'attente de réception avant qu'elle ne soit pleine. Si le programme Q Capture ne peut plus enregistrer de données sur la file d'attente d'envoi, le programme Q Capture

désactive tous les abonnements Q qui utilisent la file d'attente d'envoi ou il s'arrête, en fonction de l'option d'erreur que vous avez spécifiée pour la mappe de file d'attente de réplication.

Recommandations :

- Il est possible que vous souhaitiez que Q Apply n'applique plus de transactions à partir de la file d'attente de réception lorsqu'une erreur se produit dans les situations suivantes :
 - Les erreurs ne sont pas tolérées dans votre scénario et vous espérez qu'il ne s'en produira pas souvent.
 - Vous avez plusieurs abonnements Q définis sur des tables non associées et les données sont transportées sur plusieurs mappes de files de réplication. Vous souhaitez que tous les abonnements Q s'arrêtent lorsque l'un d'entre eux contient une erreur. Vous pouvez vérifier le problème survenu pour cet abonnement puis, après avoir résolu ce problème, redémarrer la file d'attente de réception. Cette option est adaptée si peu de changements sont répliqués vers la table cible car elle permet de préserver les transactions et de maintenir une intégrité référentielle entre les tables associées.
- Il est possible que vous ne souhaitiez pas que Q Apply arrête d'appliquer les transactions à partir de la file d'attente de réception si le résultat obtenu est trop lourd pour votre scénario lorsqu'une erreur se produit.

Chapitre 6. Configuration de la réplication Q multidirectionnelle

Avec la réplication Q, vous pouvez répliquer des données (dans un sens et dans l'autre) entre des tables sur au moins deux serveurs tandis que les applications mettent à jour les copies identiques d'une table sur tous les serveurs tout en conservant toutes les copies de la table synchronisée. Ce type de réplication est connu sous le nom de réplication *multidirectionnelle*.

Réplication bidirectionnelle

Dans une réplication bidirectionnelle, les changements effectués sur une copie d'une table sont répliqués vers une deuxième copie de cette même table et les changements de cette deuxième copie sont répliqués en retour vers la première copie.

La réplication bidirectionnelle a les caractéristiques suivantes :

- Les applications sur l'un des serveurs, quel qu'il soit, peuvent mettre à jour les mêmes lignes de ces tables simultanément. Cependant, il est peu probable, voire impossible, que des données identiques dans les tables répliquées puissent être mises à jour simultanément par les deux serveurs. Soit la même ligne est mise à jour par un serveur à un moment donné, soit un serveur met uniquement à jour certaines colonnes ou lignes de vos données et l'autre serveur met à jour différentes colonnes et lignes.
- Vous pouvez choisir quelle copie de la table prévaudra en cas de conflit.

La collecte des deux copies d'une seule table est appelée *table logique*. Chaque serveur possède une copie de la table. Chaque copie de la table :

- Doit avoir le même nombre de colonnes et de lignes
- Doit avoir les mêmes noms de colonne
- Doit avoir des types de données compatibles
- Peut avoir des noms et des schémas différents

Dans ce type de réplication, il n'est pas possible de manipuler les données en utilisant le programme Q Apply qui transmet les données à une procédure stockée. Sur chaque serveur, il y a au moins un programme Q Capture et un programme Q Apply qui fait partie d'une configuration bidirectionnelle.

Avertissement : Les tables de contrôle des programmes Q Capture et Q Apply qui sont sur chaque serveur individuel doivent avoir le même nom de schéma. Par exemple, si vous avez un serveur nommé SERVER_RED et un serveur nommé SERVER_GREEN, les programmes Q Capture et Q Apply qui sont sur le serveur SERVER_RED doivent avoir le même schéma et les programmes Q Capture et Q Apply qui sont sur le serveur SERVER_GREEN doivent avoir le même schéma.

Objets de réplication pour une réplication bidirectionnelle

Dans une configuration bidirectionnelle, vous devez avoir le nombre approprié de mappes de files de réplication et d'abonnements Q :

Nombre de mappes de files de réplication

Vous devez avoir deux mappes de files de réplication entre chaque paire

de serveurs qui entrent en jeu dans une réplication bidirectionnelle. Par exemple, si vous disposez de deux serveurs nommés SERVER_RED et SERVER_GREEN, vous devez avoir deux mappes de files de réplication :

- Une pour identifier les files d'attente WebSphere MQ qui transportent les données du serveur SERVER_RED vers le serveur SERVER_GREEN
- Une pour identifier les files d'attente de WebSphere MQ qui transportent les données du serveur SERVER_GREEN vers le serveur SERVER_RED

Nombre d'abonnements Q

Pour chaque table logique répliquée dans une réplication bidirectionnelle, vous devez avoir deux abonnements Q entre les deux serveurs. Par exemple, si vous disposez de deux serveurs ayant pour nom SERVER_RED et SERVER_GREEN, deux abonnements Q sont alors générés pour vous :

- Un abonnement de la table source sur le serveur SERVER_RED vers la table cible sur le serveur SERVER_GREEN
- Un abonnement de la table source sur le serveur SERVER_GREEN vers la table cible sur le serveur SERVER_RED

Si vous disposez de deux tables logiques, il vous faudra alors quatre abonnements Q ; pour trois tables logiques, il vous faudra six abonnements Q, etc.

La figure 15 affiche la réplication bidirectionnelle d'une table logique entre deux serveurs. Pour une table logique, vous avez besoin de deux abonnements Q et deux mappes de files de réplication.

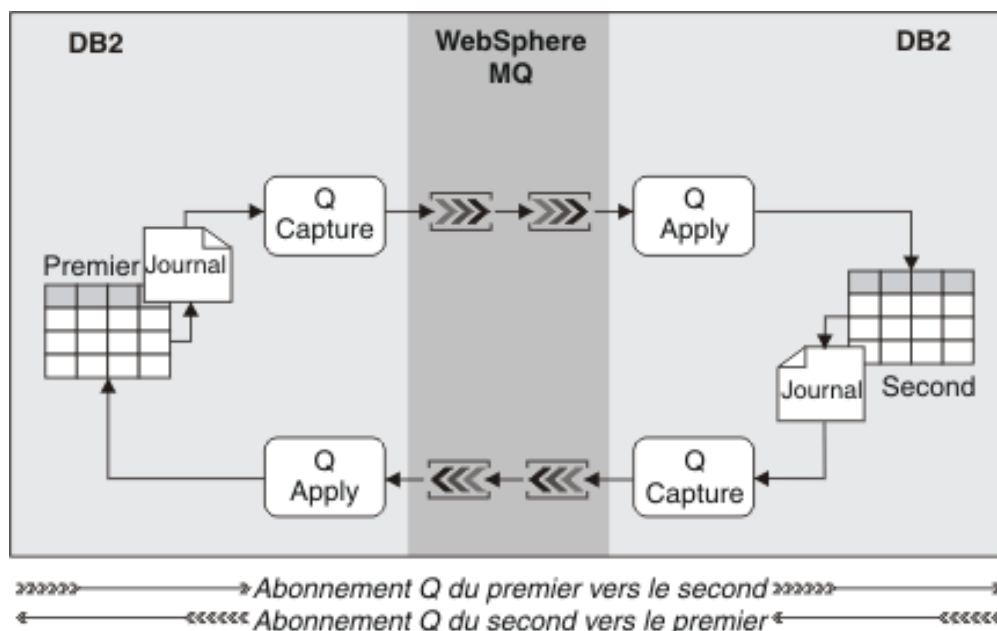


Figure 15. abonnements Q entre deux copies d'une table pour une réplication bidirectionnelle. Les changements sont répliqués depuis chaque copie de la table vers l'autre copie de cette table sur les files d'attente WebSphere MQ.

Détection de conflit dans une réplication bidirectionnelle

Dans une réplication bidirectionnelle, il est possible que des données répliquées d'une table source dans un abonnement Q entrent en conflit avec des changements effectués sur la table cible correspondante en raison d'une autre application que le programme Q Apply. La réplication bidirectionnelle utilise des valeurs de données

permettant de détecter et de résoudre des conflits. Vous pouvez choisir les valeurs de données utilisées pour détecter les conflits. Ces valeurs de données peuvent être uniquement des valeurs de colonne clés, des valeurs de colonne modifiées ou toutes les valeurs de colonne.

Par exemple, imaginez un scénario dans lequel des applications sur un système apportent des modifications aux tables d'un serveur (SERVER_RED) et que ce serveur réplique ces modifications à des tables identiques d'un serveur (SERVER_GREEN) sur un système de secours. Lors d'une défaillance du premier système, vos applications commencent à utiliser les tables sur SERVER_GREEN. Lorsque le premier système est à nouveau opérationnel, vous voulez répliquer les modifications apportées du serveur SERVER_GREEN vers SERVER_RED. Cependant, lors de la défaillance du premier système, il est possible que certaines données n'aient pas été répliquées vers le second système. Ces données, qui sont maintenant anciennes, doivent être remplacées par les données répliquées à partir de SERVER_GREEN. Lorsque vous répliquez les nouvelles données, le programme Q Apply du serveur SERVER_RED reconnaît les conflits et force l'application des changements provenant de SERVER_GREEN vers SERVER_RED.

Vous pouvez choisir la manière dont les programmes Q Apply contrôlent la présence de conflits sur les deux serveurs lorsqu'ils essaient d'appliquer des données sur les deux copies de la table et les actions qu'ils entreprennent en cas de détection de conflits. Les choix effectués en ce qui concerne les règles et les actions de conflit sont fondamentaux car ils affectent la manière dont les lignes seront appliquées.

Réplication entre homologues

Dans la réplication entre homologues (aussi appelée réplication multimaster), les mises à jour effectuées sur un serveur, quel qu'il soit, sont répliquées sur tous les autres serveurs associés.

Les caractéristiques de la réplication entre homologues sont les suivantes :

- La réplication s'effectue entre des tables sur au moins deux serveurs.
- Les applications sur l'un des serveurs, quel qu'il soit, peuvent mettre à jour les mêmes lignes et colonnes de ces tables simultanément.
- Tous les serveurs sont des homologues de même niveau, avec le même niveau de propriété des données, c'est-à-dire qu'il n'y a pas de serveur "maître" ou de propriétaire source des données.

Dans une réplication entre homologues, vous répliquez des copies des tables entre plusieurs serveurs. La collecte de toutes les copies d'une seule table est appelée table logique. Chaque serveur possède une copie de la table. Chaque copie de la table :

- Doit avoir le même nombre de colonnes et de lignes
- Doit avoir les mêmes noms de colonne
- Doit avoir des types de données compatibles
- Peut avoir des noms et des schémas différents

Dans la réplication entre homologues, la convergence des données est assurée entre toutes les copies de la table logique, c'est-à-dire que chaque copie de cette table aura finalement un état identique à celui des autres copies et aura pour valeurs les valeurs validées les plus récentes. La réplication entre homologues étant asynchrone, il est possible que les copies des tables ne convergent pas jusqu'à ce

vos applications n'effectuent plus de modifications, que ces modifications soient répliquées et que tous les messages soient traités.

Dans ce type de réplication, il n'est pas possible de manipuler les données en utilisant le programme Q Apply qui transmet les données à une procédure stockée. Sur chaque serveur, il y a au moins un programme Q Capture et un programme Q Apply faisant partie d'une configuration entre homologues.

Important : les tables de contrôle des programmes Q Capture et Q Apply qui sont sur chaque serveur doivent avoir le même nom de schéma. Par exemple, si vous avez un serveur nommé SERVER_RED et un serveur nommé SERVER_GREEN, les programmes Q Capture et Q Apply qui sont sur le serveur SERVER_RED doivent avoir le même schéma et les programmes Q Capture et Q Apply qui sont sur le serveur SERVER_GREEN doivent avoir le même schéma.

Dans une configuration entre homologues, l'identification et la résolution de conflits est gérée automatiquement par Q Apply et ce d'une manière qui garantit la convergence des données. Il n'est donc pas nécessaire de configurer quoi que ce soit pour identifier et résoudre les conflits. La réplication Q gère des informations supplémentaires pour le suivi de la version de chaque modification des données et le système de réplication Q utilise ces informations de gestion des versions supplémentaires pour identifier et résoudre les conflits.

Toutes les tables qui sont répliquées dans une réplication entre homologues sont modifiées afin d'inclure deux colonnes qui sont uniquement utilisées par la réplication Q : une colonne référence temporelle et une colonne chiffre entier. Elles sont gérées par des déclencheurs. Ces colonnes supplémentaires et ces déclencheurs sont créés lorsque vous créez les abonnements Q de réplication entre homologues. Les colonnes de gestion des versions indiquent la version de la ligne la plus actuelle. En examinant les valeurs de ces colonnes, il est possible de déterminer à quel moment la dernière mise à jour de la ligne a eu lieu et quel serveur l'a effectuée.

L'identification et la résolution de conflits se base sur les contenus de ces colonnes. Si un conflit est identifié, c'est la version de la ligne la plus récente qui est conservée et qui est celle contenant la valeur de référence temporelle la plus récente (après correction des heures pour les fuseaux horaire)

Lorsque vous créez un abonnement Q pour une réplication entre homologues avec le programme de ligne de commande ASNCLP ou le Centre de réplication, les options suivantes de traitement des conflits sont automatiquement paramétrées dans la table IBMQREP_TARGETS table :

règles concernant les conflits

V (vérification de la version) : Q Apply vérifie la colonne des versions avant d'appliquer une ligne.

action à prendre en cas de conflit

F (force) : Q Apply tente forcer les modifications. Il faut pour cela que Q Capture envoie toutes les colonnes. La valeur CHANGED_COLS_ONLY doit donc être paramétrée sur N (non) dans la table IBMQREP_SUBS

Avertissement : La règle V concernant les conflits et la règle F d'action à prendre en cas de conflit sont obligatoires pour une réplication entre homologues. Ne changez pas ces paramètres dans les tables de contrôle

Les colonnes de gestion des versions servant à identifier les conflits dans une réplication entre homologues, les colonnes contenant les types de données objet LOB sont traitées comme les colonnes contenant d'autres types de données.

Les rubriques suivantes décrivent le nombre de maps de files de réplication et d'abonnements Q nécessaires dans une réplication entre homologues et comment cette réplication entre homologues supporte l'intégrité référentielle :

- «Objets de réplication pour une réplication entre homologues avec deux serveurs»
- «Objets de réplication pour une réplication entre homologues avec trois serveurs ou plus», à la page 102
- «Résolution de conflits et intégrité référentielle», à la page 103

Objets de réplication pour une réplication entre homologues avec deux serveurs

Dans une configuration entre homologues, vous devez avoir le nombre approprié de maps de files de réplication et d'abonnements Q :

Nombre de maps de files de réplication

Vous devez avoir deux maps de files de réplication entre chaque paire de serveurs qui entrent en jeu dans la réplication entre homologues. Par exemple, si vous disposez de deux serveurs nommés SERVER_RED et SERVER_GREEN, vous devez avoir deux maps de files de réplication :

- Une pour identifier les files d'attente WebSphere MQ qui transportent les données du serveur SERVER_RED vers le serveur SERVER_GREEN
- Une pour identifier les files d'attente de WebSphere MQ qui transportent les données du serveur SERVER_GREEN vers le serveur SERVER_RED

Nombre d'abonnements Q

Pour chaque table logique répliquée dans une réplication entre homologues, vous devez avoir deux abonnements Q entre les deux serveurs. Par exemple, si vous disposez de deux serveurs ayant pour nom SERVER_RED et SERVER_GREEN, deux abonnements Q sont alors générés pour vous :

- Un abonnement de la table source sur le serveur SERVER_RED vers la table cible sur le serveur SERVER_GREEN
- Un abonnement de la table source sur le serveur SERVER_GREEN vers la table cible sur le serveur SERVER_RED

Si vous disposez de deux tables logiques, il vous faudra alors quatre abonnements Q ; pour trois tables logiques, il vous faudra six abonnements Q, etc.

La figure 16, à la page 102 affiche la réplication entre homologues d'une table logique entre deux serveurs. Pour une table logique répliquée entre deux serveurs, il vous faut deux abonnements Q : un abonnement pour répliquer les données de la table A sur la table B et un abonnement pour répliquer les données de la table B sur la table A. Il vous faut également au moins deux maps de files de réplication.

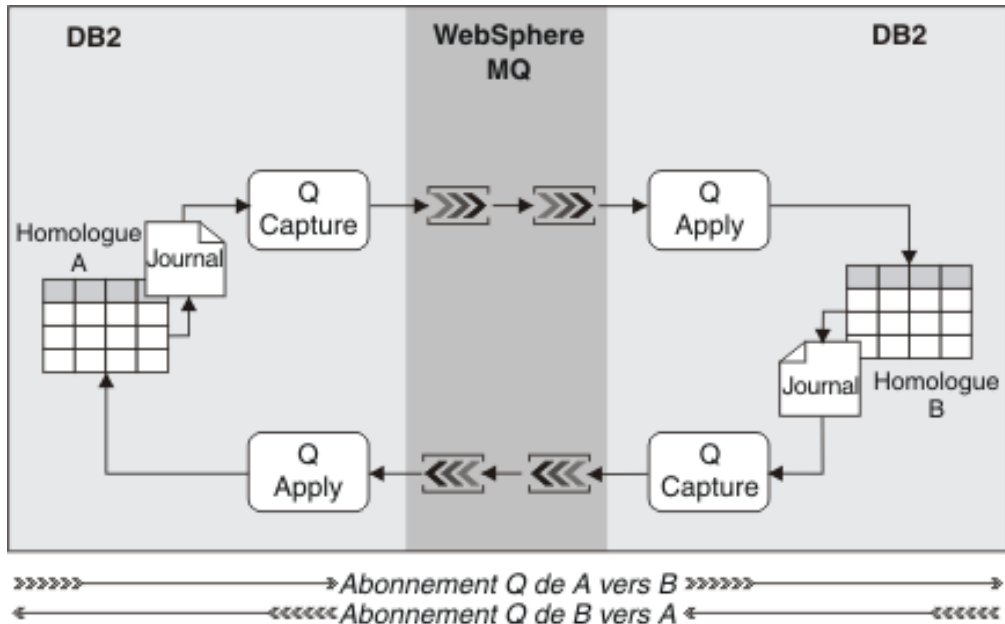


Figure 16. Abonnements Q dans la réplication entre homologues avec deux serveurs. Les changements sont répliqués depuis chaque copie de la table vers l'autre copie de cette table sur les files d'attente WebSphere MQ.

Objets de réplication pour une réplication entre homologues avec trois serveurs ou plus

Dans une configuration entre homologues comportant trois serveurs ou plus, vous devez avoir le nombre approprié de mappes de files de réplication et d'abonnements Q :

Nombre de mappes de files de réplication

Vous devez avoir deux mappes de files de réplication entre chaque paire de serveurs qui entrent en jeu dans la réplication entre homologues. Vous pouvez calculer le nombre de mappes de files de réplication dont vous avez besoin à l'aide de l'équation $n*(n-1)$, où n correspond au nombre de serveurs. Si, par exemple, vous disposez de trois serveurs ayant pour nom SERVER_RED, SERVER_BLUE et SERVER_GREEN, il vous faut six mappes de files de réplication :

- Une pour identifier les files d'attente WebSphere MQ qui transportent les données du serveur SERVER_RED vers le serveur SERVER_GREEN
- Une pour identifier les files d'attente de WebSphere MQ qui transportent les données du serveur SERVER_GREEN vers le serveur SERVER_RED
- Une mappe pour identifier les files d'attente de WebSphere MQ qui transportent les données du serveur SERVER_RED au serveur SERVER_BLUE
- Une mappe pour identifier les files d'attente de WebSphere MQ qui transportent les données du serveur SERVER_BLUE au serveur SERVER_RED
- Une mappe pour identifier les files d'attente de WebSphere MQ qui transportent les données du serveur SERVER_BLUE au serveur SERVER_GREEN
- Une mappe pour identifier les files d'attente de WebSphere MQ qui transportent les données du serveur SERVER_GREEN au serveur SERVER_BLUE

Nombre d'abonnements Q

Pour chaque table logique répliquée dans une réplication entre homologues, il y a deux abonnements Q entre les deux serveurs. Vous pouvez calculer le nombre d'abonnements Q dont vous avez besoin à l'aide de l'équation $n*(n-1)$, où n correspond au nombre de serveurs. Si vous disposez par exemple de trois serveurs ayant pour nom SERVER_RED, SERVER_GREEN et SERVER_BLUE, six abonnements Q sont alors générés pour vous :

- Un abonnement de la table source sur le serveur SERVER_RED vers la table cible sur le serveur SERVER_GREEN
- Un abonnement de la table source sur SERVER_GREEN vers la table cible sur SERVER_RED
- Un abonnement de la table source sur SERVER_RED vers la table cible sur SERVER_BLUE
- Un abonnement de la table source sur SERVER_BLUE vers la table cible sur SERVER_RED
- Un abonnement de la table source sur SERVER_BLUE vers la table cible sur SERVER_GREEN
- Un abonnement de la table source sur SERVER_GREEN vers la table cible sur SERVER_BLUE

Si vous disposez de deux tables logiques, il vous faudra alors 12 abonnements Q ; pour trois tables logiques, il vous faudra 18 abonnements Q etc.

La figure 17 affiche la réplication entre homologues d'une table logique entre trois serveurs. Dans ce cas, il vous faut six abonnements Q, soient deux abonnements entre chaque paire de serveurs. Il vous faut également au moins six mappes de files de réplication.

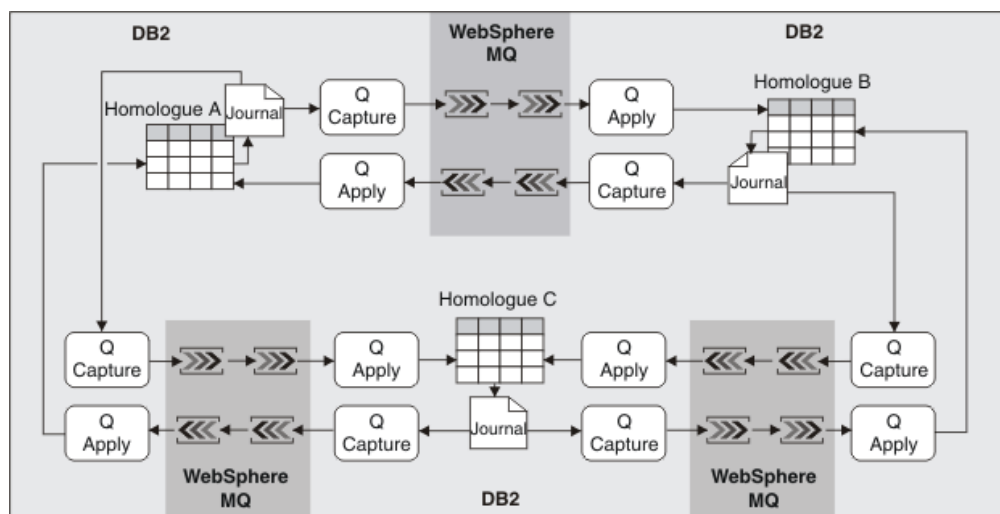


Figure 17. Abonnements Q dans la réplication entre homologues avec trois serveurs. Les modifications sont répliquées depuis chaque copie de la table vers l'autre copie de cette table sur les files d'attente de WebSphere MQ.

Résolution de conflits et intégrité référentielle

Dans presque tous les cas, la réplication entre homologues garantit la convergence vers le même état de toutes les copies d'une table répliquée, même lorsque des modifications conflictuelles ont lieu sur différentes copies. Cependant, des conflits

non résolubles peuvent survenir lorsqu'un conflit provient de valeurs répliquées dans des contraintes d'unicité définies dans des colonnes autres que les colonnes clés ou lorsqu'un conflit provient de violations de contrainte référentielle. Lorsqu'un conflit non résoluble survient, la ligne touchée est enregistrée dans la table `IBMQREP_EXCEPTIONS` et Q Apply exécute les actions à effectuer en cas de conflit que vous avez spécifiées pour l'abonnement Q.

Si vous souhaitez que les conflits particuliers, non résolubles soient tolérés et que Q Apply n'exécute pas les actions à effectuer en cas de conflit que vous avez spécifiées pour l'abonnement Q, vous pouvez alors préciser des valeurs `SQLSTATE` acceptables en paramétrant les états `OKSQLSTATES` pour l'abonnement Q. Cependant, même si vous précisez des états SQL spécifiques dans `OKSQLSTATES`, la réplication entre homologues ne garantira toujours pas la convergence de toutes les copies de la table en ce qui concerne les conflits résultant de violations de contrainte référentielle ou de valeurs répliquées dans les contraintes d'unicité définies dans les colonnes autres que les colonnes clés. Vous pouvez utiliser la fonctionnalité d'analyse des différences de tables et la fonctionnalité de correction de tables pour rechercher et corriger les différences causées par n'importe quel conflit non résoluble que vous tolérez.

Il n'est pas possible de résoudre les conflits lorsque des modifications surviennent sur différentes copies de la table répliquée qui introduisent la même valeur pour une contrainte d'unicité sur les colonnes autres que les colonnes clés dans l'abonnement Q. Si vous précisez que l'état `SQLSTATE 23505` est autorisé en ajoutant la valeur au `OKSQLSTATES` pour l'abonnement Q, aucun conflit clé unique non résoluble n'entraîne alors l'exécution par Q Apply des actions à effectuer en cas de conflit spécifiées pour l'abonnement Q.

Il n'est pas possible de résoudre les conflits lorsque des modifications surviennent dans des lignes de différentes copies de tables sur lesquelles des contraintes référentielles sont définies. Ces conflits sont dus soit à des retards dans la propagation des messages relatifs à ces lignes, soit à de véritables conflits. On parle de véritable conflit, par exemple, lorsqu'une ligne parente est supprimée dans une copie et qu'une ligne enfant est insérée en même temps dans une autre copie. Lorsque Q Apply tente d'insérer la ligne enfant dans la copie dans laquelle la ligne parente a été supprimée, cela crée un conflit non résoluble et Q Apply enregistre alors la ligne enfant dans la table `IBMQREP_EXCEPTIONS` avec l'état `SQLSTATE 23503`. Lorsque Q Apply tente de supprimer la ligne parente dans la copie dans laquelle la ligne enfant a été insérée, la suppression échoue si la règle de suppression concernant la contrainte référentielle est de limiter les suppressions (`DELETE RESTRICT`). Q Apply enregistre alors la ligne parente dans la table `IBMQREP_EXCEPTIONS` avec l'état `SQLSTATE 23504` ou `SQLSTATE 23001`.

L'insertion et la suppression d'une ligne enfant simultanée par la règle de suppression (`CASCADE DELETE`) concernant l'intégrité référentielle au moment où une ligne parente est supprimée, est un autre exemple de conflit véritable. Dans ce cas, lorsque la suppression en cascade (`cascade delete`) de la ligne enfant est répliquée vers d'autres copies de la table, il est possible que ces copies ne trouvent pas la ligne enfant et un état `SQLSTATE 02000` est alors rapporté. Il est possible que cet état `SQLSTATE 02000` soit dû à des retards dans la propagation des messages relatifs aux lignes. L'insertion d'une ligne enfant dans la COPIE 2 peut arriver à la COPIE 3 avant que l'insertion de la ligne parente dans la COPIE 1 n'arrive à la COPIE 3.

Intégrité référentielle pour les bases de données partitionnées

Dans un environnement de bases de données multiples partitionnées avec des tables possédant des relations d'intégrité référentielle, assurez-vous que le parent et l'enfant se trouvent dans la même partition. Si le parent et l'enfant ont une relation d'intégrité référentielle et qu'ils ne figurent pas dans la même partition, des problèmes d'intégrité référentielle peuvent se déclarer au niveau de la cible avec pour résultat les états SQLSTATE 23504, 23001 ou 23503 (qui correspondent aux codes SQLCODE 530 et 532).

Eviter les blocages dans la table IBMQREP_DELTOMB

Q Apply utilise la table IBMQREP_DELTOMB pour enregistrer les suppressions conflictuelles dans une réplication entre homologues. Si vous rencontrez des problèmes de blocage dans cette table de contrôle sur l'un des serveurs, augmentez la valeur du paramètre **deadlock_retries** de Q Apply. Essayez aussi, si possible, de réduire les suppressions conflictuelles dans votre charge de travail.

Réplication bidirectionnelle et réplication entre homologues

Si vous voulez répliquer des données entre des tables de deux serveurs, vous avez le choix entre deux types de réplication multidirectionnelle : la réplication bidirectionnelle et la réplication entre homologues.

Les informations suivantes vous permettront de choisir entre réplication bidirectionnelle et réplication entre homologues pour satisfaire au mieux vos besoins métier. S'il vous faut plus de deux serveurs pour votre configuration, la réplication entre homologues est alors la seule solution qui s'offre à vous en termes de réplication multidirectionnelle.

Scénarios fonctionnant le mieux avec la réplication bidirectionnelle

Envisagez la réplication bidirectionnelle pour les raisons suivantes :

- Vous ne pensez pas que des conflits puissent avoir lieu dans votre configuration et vous n'avez pas besoin de vérifier s'il en existe. Pour un trafic temps système et réseau minimum, précisez pour les deux serveurs d'ignorer les conflits.
- Vous ne pensez pas que des conflits puissent avoir lieu dans votre configuration mais, par mesure de sécurité, vous souhaitez le vérifier et vous considérez comme acceptable qu'un serveur triomphe sur l'autre si une collision imprévue de données a lieu.
- Un des serveurs met à jour certaines colonnes de vos données et l'autre met à jour les colonnes restantes. Si vous précisez que le programme Q Apply doit vérifier s'il existe des conflits concernant les colonnes clés et les colonnes modifiées, ce programme fusionne les mises à jour des différentes colonnes de la même ligne.
- Un des serveurs met à jour certaines colonnes de vos données et l'autre met à jour les colonnes restantes.

Scénarios fonctionnant le mieux avec la réplication entre homologues

Envisagez plutôt la réplication entre homologues s'il est possible que des conflits apparaissent dans les colonnes contenant les types de données objet LOB. Les colonnes de gestion des versions servant à identifier les conflits dans une

réplication entre homologues, les colonnes contenant les types de données objet LOB sont traitées comme les colonnes contenant d'autres types de données.

Création d'abonnements Q pour une réplication bidirectionnelle

Vous pouvez créer des abonnements Q qui spécifient les changements à enregistrer à partir de l'une ou l'autre des deux tables, les files d'attente à utiliser pour l'échange de messages sur ces changements et comment traiter ces messages. Les changements dans l'une des deux tables sont répliqués vers l'autre table.

Avant de commencer

- Planifiez la manière dont vous voulez regrouper les mappes de files de réplication et les abonnements Q.
- Sur le serveur comportant la première copie de la table, créez les tables de contrôle pour les programmes Q Capture et Q Apply. Les tables de contrôle des programmes qui sont sur chaque serveur doivent comporter le même schéma.
- Sur le serveur comportant la deuxième copie de la table, créez les tables de contrôle pour les programmes Q Capture et Q Apply.
- Créez les deux mappes de files de réplication qui transporteront les données entre chaque serveur. Il vous faut une mappe de files de réplication pour répliquer les données de la première copie de la table vers la seconde et une mappe de files de réplication pour répliquer en retour les données de la deuxième copie vers la première. (Vous pouvez effectuer cette tâche avant ou pendant la création des abonnements Q.)

Restrictions

- Les procédures stockées ne peuvent pas entrer dans une réplication bidirectionnelle.
- Les valeurs antérieures des colonnes objet LOB n'étant pas répliquées avec une réplication bidirectionnelle, les conflits concernant ces colonnes ne sont pas identifiés.
- Les colonnes IDENTITY dans la table cible ne peuvent pas être définies comme GENERATED ALWAYS.
- Les abonnements Q des tables ayant des relations d'intégrité référentielle les unes avec les autres, doivent être créés simultanément (dans la même commande CREATE QSUB lorsque vous utilisez le programme de ligne de commande ASNCLP ou dans la même session avec l'assistant Créer des abonnements Q dans le Centre de réplication).

A propos de cette tâche

Un abonnement Q est créé pour répliquer les transactions de la première copie de la table vers la deuxième copie et un autre abonnement est créé pour répliquer en retour les transactions de cette deuxième copie vers la première. Lorsque vous créez des abonnements Q pour une réplication bidirectionnelle à l'aide du programme de ligne de commande ASNCLP ou du Centre de réplication, l'outil d'administration crée ces deux abonnements en une seule fois.

Procédure

Pour créer des abonnements Q pour une réplication bidirectionnelle, appliquez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande CREATE QSUB pour la réplication bidirectionnelle. Par exemple, les commandes suivantes définissent l'environnement puis créent deux abonnements Q bidirectionnels pour la table EMPLOYEE sur les serveurs SAMPLE et SAMPLE2 :</p> <pre>SET SUBGROUP "bidirgroup"; SET BIDI NODE 1 SERVER DBALIAS SAMPLE SCHEMA RED; SET BIDI NODE 2 SERVER DBALIAS SAMPLE2 SCHEMA BLUE; SET CONNECTION SOURCE SAMPLE.RED TARGET "SAMPLE2".BLUE REPLQMAP "SAMPLE_RED_TO_SAMPLE2_BLUE"; SET CONNECTION SOURCE SAMPLE2.BLUE TARGET SAMPLE.RED REPLQMAP "SAMPLE2_BLUE_TO_SAMPLE_RED"; SET TABLES (SAMPLE.RED.RED.EMPLOYEE); CREATE QSUB SUBTYPE B FROM NODE SAMPLE.RED SOURCE ALL CHANGED ROWS Y HAS LOAD PHASE I TARGET CONFLICT RULE C CONFLICT ACTION F FROM NODE SAMPLE2.BLUE SOURCE ALL CHANGED ROWS N HAS LOAD PHASE E TARGET CONFLICT RULE C CONFLICT ACTION I;</pre> <p>Les instructions SET CONNECTION spécifient les deux mappes de files de réplication qui sont utilisés. Les instructions FROM NODE spécifient les options qui sont uniques à chaque abonnement Q.</p>
Centre de réplication	<p>Utilisez l'assistant Créer des abonnements Q. Pour ouvrir l'assistant, développez le schéma Q Capture ou Q Apply approprié, cliquez avec le bouton droit de la souris sur le dossier Abonnements Q puis sélectionnez Créer.</p> <p>Sur la page Tables cible, examinez le profil objet cible. Modifiez ce profil si nécessaire afin que les tables cible pour les abonnements Q répondent à vos besoins.</p> <p>Le profil objet cible détermine si une table cible existante est utilisée ou si une nouvelle est créée. Le Centre de réplication recherche un objet qui corresponde au système de dénomination défini dans le profil. S'il n'existe aucun objet correspondant, il en est créé un.</p>

Amélioration des performances de la réplication bidirectionnelle grâce à la table IBMQREP_IGNTRAN

Vous pouvez vous servir de la capacité du programme Q Capture à ignorer les transactions spécifiées pour améliorer les performances dans une configuration bidirectionnelle pure à deux serveurs.

A propos de cette tâche

Pour éviter la recapture des transactions, le programme Q Apply insère par défaut des signaux P2PNORECAPTURE dans la table IBMQREP_SIGNAL pour chacune des transactions qu'elle valide. Les signaux sont insérés au niveau du serveur que Q Capture partage avec Q Apply. Lorsque Q Capture lit les signaux dans le journal, il ignore ces transactions.

Lorsque les abonnements Q bidirectionnels sont nombreux, le nombre d'insertions de signaux peut affecter les performances de la réplication. Pour prévenir cela, vous pouvez spécifier que les programmes utilisent la table IBMQREP_IGNTRAN pour éviter la recapture. Cette méthode indique au programme Q Capture d'ignorer automatiquement toute modification en provenance du programme Q Apply. Par ailleurs, vous démarrerez le programme Q Apply en utilisant le paramètre de démarrage **insert_bidi_signal=N**.

Suivez les instructions ci-après pour déterminer la méthode à utiliser pour éviter la recapture :

Tableau 9. Méthodes pour éviter la recapture pour différentes configurations de réplication bidirectionnelle

Configuration	Méthode pour éviter la recapture
<p>Multiples configurations bidirectionnelles entre deux serveurs ou combinaison de configurations bidirectionnelles, unidirectionnelles et entre homologues</p>	<p>Vous devez accepter la méthode par défaut</p> <p>La méthode d'insertion de signaux par défaut assure que toutes les modifications se propagent correctement à travers les serveurs.</p> <p>Si vous commencez avec une topologie bidirectionnelle pure à deux serveurs et envisagez d'ajouter ultérieurement un branchement de topologie unidirectionnelle ou entre homologues, il vous faudra aussi accepter la méthode pour éviter la recapture par défaut.</p>
<p>Configuration bidirectionnelle pure à deux serveurs</p>	<p>Les performances peuvent être améliorées en utilisant la table IBMQREP_IGNTRAN pour éviter la recapture</p> <p>Si vous choisissez la méthode de la table IBMQREP_IGNTRAN, n'ajoutez pas par la suite de branchement de configurations unidirectionnelle ou entre homologues à la configuration bidirectionnelle.</p>

Procédure

Pour utiliser la table IBMQREP_IGNTRAN et éviter ainsi la recapture dans la réplication bidirectionnelle :

1. Insérez un identificateur du programme Q Apply dans la table IBMQREP_IGNTRAN au niveau du serveur que se partagent les programmes Q Apply et Q Capture dans la configuration bidirectionnelle. En fonction de votre système d'exploitation, utilisez les instructions SQL suivantes :

z/OS

```
insert into schéma.IBMQREP_IGNTRAN (
    PLANNAME,
    IGNTRANTRC
) values (
    'plan_qapply',
    'N' );
```

Où *schéma* est le nom du schéma partagé par les programmes Q Apply et Q Capture Apply au niveau du serveur et *plan_qapply* est le nom de

plan pour le programme Q Apply, par exemple ASNQA910 pour un programme Q Apply de version 9.1.

Linux UNIX Windows

```
insert into schéma.IBMQREP_IGNTRAN (  
    AUTHID,  
    IGNTRANTRC  
) values (  
    'ID_auth_qapply',  
    'N');
```

Où *schéma* est le nom du schéma partagé par les programmes Q Apply et Q Capture Apply au niveau du serveur et *ID_auth_qapply* est l'identificateur utilisateur de démarrage du programme Q Apply.

Si vous utilisez l'option de la table IBMQREP_IGNTRAN, l'identificateur utilisateur du programme Q Apply devra être unique et n'être pas partagé par d'autres applications, si ce n'est le programme Q Capture. Autrement, le programme Q Capture ignorera par erreur également les transactions issues de ces autres applications. Sous z/OS, ces cas sont rares car les noms de plan sont uniques. Cependant, sous Linux, UNIX et Windows, il n'est pas inhabituel d'exécuter le programme Q Apply sous le même identificateur utilisateur que d'autres applications.

Si vous avez effectué une migration de Q Apply vers la version 10.1 sous z/OS : Le nom de plan Q Apply pour la version 10.1 sous z/OS a changé pour ASNQA101 (anciennement ASNQA910). Pour vérifier que des transactions du programme Q Apply version 10.1 sont ignorées, une fois la migration de Q Apply à la version 10.1 terminée, entrez le nouveau nom de plan ASNQA101 dans la table IBMQREP_IGNTRAN, puis réinitialisez le programme Q Capture ou arrêtez et redémarrez Q Capture pour qu'il puisse lire le nouveau nom de plan. A titre de protection, vous pouvez placer les noms de plan d'ASNQA101 et d'ASNQA910 dans la table IBMQREP_IGNTRAN.

Le programme Q Capture ignorera toutes les transactions répliquées en provenance du nom de plan ou de l'identificateur utilisateur Q Apply indiqué, mais continuera de lire et de traiter les signaux de Q Apply dans la table IBMQREP_SIGNAL.

2. Lorsque vous démarrez le programme Q Apply, définissez le paramètre de démarrage **insert_bidi_signal**=n. La syntaxe de formulaire courte est **INS=N**.
3. Facultatif : Pour améliorer les performances lorsque vous utilisez le paramètre **insert_bidi_signal**=n, veillez à ce que la valeur de la colonne IGNTRANTRC dans la table IBMQREP_IGNTRANTRC soit définie sur N (pas de suivi). Une valeur de N, la valeur par défaut, empêche le programme Q Capture d'insérer une ligne dans la table IBMQREP_IGNTRANTRC pour chaque transaction qu'il ne recapture pas, et réduit les frais de maintenance sur la table.

Options de détection de conflit (réplication bidirectionnelle)

Les choix que vous faites en matière de règles de conflit et d'actions de conflit affectent la manière dont les lignes seront appliquées. Les règles de conflit déterminent le nombre de données vérifiées permettant de détecter un conflit et les types de conflit détectés. Si vous choisissez de vérifier plus de données pour détecter les conflits, le programme Q Capture doit envoyer plus de données au programme Q Apply pour rendre possible la vérification de ces données, ce qui peut influencer les performances et le trafic sur le réseau.

Pour la détection de conflit dans une réplication bidirectionnelle, les valeurs antérieures au niveau du serveur source sont comparées avec les valeurs actuelles au niveau du serveur cible. En fonction du niveau de détection de conflit que vous choisissez, le programme Q Capture envoie une combinaison différente de valeurs antérieures et postérieures au programme Q Apply. Les informations fournies ici sont destinées à vous permettre de prendre des décisions plus éclairées en ce qui concerne le niveau de détection de conflit.

Remarque : Dans la mesure où les valeurs antérieures sont utilisées pour détecter des conflits dans une réplication bidirectionnelle et que la réplication Q ne réplique pas les valeurs antérieures pour les types de données LOB, les conflits dans les colonnes LOB ne sont pas détectés.

Les sections suivantes décrivent vos options en ce qui concerne la détection de conflit dans une réplication bidirectionnelle et les résultats des différentes combinaisons des options de conflit :

- «Comment le programme Q Apply vérifie la présence de conflits ?»
- «Comment les conflits sont-ils résolus sur chaque serveur ?», à la page 111
- «Résultats des différents choix de contrôle et de résolution des conflits», à la page 111

Comment le programme Q Apply vérifie la présence de conflits ?

Vous pouvez faire en sorte que le programme Q Apply vérifie un des groupes de colonnes suivants lors de la détermination des conflits :

- Uniquement les colonnes de clé
- Colonnes de clé et colonnes modifiées
- Toutes les colonnes

Uniquement les colonnes de clé

Le programme Q Apply essaie de mettre à jour ou de supprimer la ligne cible en vérifiant les valeurs des colonnes de clé. Le programme Q Apply détecte les conflits suivants :

- Une ligne est introuvable dans la table cible.
- Une ligne est un doublon d'une ligne qui existe déjà dans la table cible.

Avec cette règle de conflit, le programme Q Capture envoie la quantité de données la plus faible au programme Q Apply pour le contrôle de conflit. Aucune valeur antérieure n'est envoyée et seules les valeurs postérieures des colonnes modifiées sont envoyées.

Colonnes de clé et colonnes modifiées

Le programme Q Apply essaie de mettre à jour ou de supprimer la ligne cible en vérifiant les valeurs des colonnes de clé et les colonnes modifiées au cours de la mise à jour. Le programme Q Apply détecte les conflits suivants :

- Une ligne est introuvable dans la table cible.
- Une ligne est un doublon d'une ligne qui existe déjà dans la table cible.
- Une ligne est mise à jour sur les deux serveurs simultanément et les mêmes valeurs de colonne ont été modifiées.

Si une ligne est mise à jour sur deux serveurs simultanément et que les différentes valeurs de colonne ont été modifiées, il n'y a pas de conflit.

Avec cette règle de conflit, le programme Q Apply fusionne les mises à jour qui affectent les différentes colonnes dans la même ligne. Puisque le programme Q Apply a besoin des valeurs antérieures des colonnes modifiées pour cette action de conflit, le programme Q Capture envoie les valeurs antérieures des colonnes modifiées.

Toutes les colonnes

Le programme Q Apply essaie de mettre à jour ou de supprimer la ligne cible en vérifiant toutes les colonnes de la table cible. Avec cette règle de conflit, le programme Q Capture envoie le plus gros volume de données au programme Q Apply pour le contrôle de conflit.

Comment les conflits sont-ils résolus sur chaque serveur ?

Pour chaque serveur, vous pouvez choisir quelle action celui-ci effectue lorsqu'un conflit se produit. Chaque serveur peut soit forcer la ligne en conflit soit ignorer le conflit. Ces options permettant de forcer et d'ignorer les conflits peuvent être couplées de deux manières différentes afin de fournir différents comportements au programme Q Apply.

Un serveur force les conflits, l'autre serveur les ignore

Un serveur (celui dont l'action de conflit consiste à ignorer le conflit) prédomine si un conflit se produit ; ce serveur est le "maître" ou propriétaire source des données. Si une ligne est mise à jour sur les deux serveurs simultanément et que les mêmes valeurs de colonne ont été modifiées, alors le serveur maître (celui dont l'action de conflit consiste à ignorer le conflit) ignore le conflit et la ligne provenant du serveur maître est forcée dans la table cible de l'autre serveur (celui dont l'action de conflit consiste à forcer le conflit). Pour cette action de conflit, le programme Q Capture envoie les valeurs antérieures de toutes les colonnes au programme Q Apply. Le programme Q Apply enregistre tous les conflits dans la table IBMQREP_EXCEPTIONS.

Les deux serveurs ignorent les conflits

Chaque fois qu'un conflit se produit en raison d'une ligne introuvable ou d'une ligne en doublon dans la table cible, le programme Q Apply enregistre le conflit dans la table IBMQREP_EXCEPTIONS, sinon le conflit est ignoré. Cette action de conflit spécifie que le programme Q Capture n'envoie pas les valeurs antérieures au programme Q Apply pour le contrôle de conflit. Seules les valeurs postérieures des colonnes modifiées sont envoyées.

Recommandation : configurez les deux serveurs de telle manière qu'ils ignorent les conflits si vous ne vous attendez pas à ce que des conflits se produisent entre les deux serveurs et si vous voulez que le temps système utilisé par les programmes Q Capture et Q Apply soit le plus faible possible.

Résultats des différents choix de contrôle et de résolution des conflits

Le tableau 10, à la page 112 décrit les résultats des différentes combinaisons d'options que vous pouvez choisir pour la détection de conflit. Dans tous les cas, le premier serveur est le serveur à partir duquel vous avez ouvert l'assistant.

Tableau 10. Résultats des différentes combinaisons d'options pour la détection de conflit

Comment contrôler la présence de conflits ?	Comment résoudre les conflits ?	Résultat
<p>Contrôler toutes les colonnes pour détecter la présence de conflits.</p>	<p>Le premier serveur est prioritaire.</p>	<p>Pour l'abonnement Q du premier serveur vers le second serveur : si un changement effectué sur la table source du premier serveur entre en conflit avec des données de la table cible du second serveur, le programme Q Apply applique la modification effectuée sur la table source à la table cible. Le programme Q Apply consigne le conflit dans la table IBMQREP_EXCEPTIONS, supprime la ligne en conflit dans la table cible et insère la ligne à partir de la table source.</p> <hr/> <p>Pour l'abonnement Q du second serveur vers le premier serveur : si un changement effectué sur la table source entre en conflit avec des données de la table cible, le programme Q Apply enregistre le conflit mais ne force pas l'application du changement dans la table cible.</p>
	<p>Le second serveur est prioritaire.</p>	<p>Pour l'abonnement Q du premier serveur vers le second serveur : si un changement effectué sur la table source entre en conflit avec des données de la table cible, le programme Q Apply enregistre le conflit mais ne force pas l'application du changement dans la table cible.</p> <hr/> <p>Pour l'abonnement Q du second serveur vers le premier serveur : si un changement effectué sur la table source entre en conflit avec des données de la table cible, le programme Q Apply applique la modification effectuée sur la table source à la table cible. Le programme Q Apply supprime la ligne en conflit dans la table cible et insère la ligne à partir de la table source. Si une suppression est en conflit et que la ligne est introuvable dans la table cible, le programme Q Apply ignore la suppression à partir de la table source.</p>
	<p>Aucun serveur n'a la priorité.</p>	<p>Le programme Q Apply enregistre tous les conflits dans la table IBMQREP_EXCEPTIONS et poursuit le traitement. Avec le temps, les deux copies d'une table logique différeront.</p>

Tableau 10. Résultats des différentes combinaisons d'options pour la détection de conflit (suite)

Comment contrôler la présence de conflits ?	Comment résoudre les conflits ?	Résultat
<p>Contrôlez uniquement la présence de conflits dans les colonnes non clés modifiées.</p>	<p>Le premier serveur est prioritaire.</p>	<p>Pour l'abonnement Q du premier serveur vers le second serveur : si un changement effectué sur une colonne non clé dans la table source entre en conflit avec un changement effectué sur la colonne correspondante de la table cible, le programme Q Apply applique dans tous les cas le changement de la table source à la table cible. Le programme Q Apply supprime la ligne en conflit dans la table cible et insère la ligne à partir de la table source. Si une suppression est en conflit et que la ligne est introuvable dans la table cible, le programme Q Apply ignore la suppression à partir de la table source.</p> <p>Pour l'abonnement Q du second serveur vers le premier serveur : si un changement effectué sur une colonne non clé dans la table source entre en conflit avec un changement effectué sur la colonne correspondante de la table cible, le programme Q Apply enregistre le conflit mais ne force pas l'application du changement dans la table cible.</p>
	<p>Le second serveur est prioritaire.</p>	<p>Pour l'abonnement Q du premier serveur vers le second serveur : si un changement effectué sur une colonne non clé dans la table source entre en conflit avec un changement effectué sur la colonne correspondante de la table cible, le programme Q Apply enregistre le conflit mais ne force pas l'application du changement dans la table cible.</p> <p>Pour l'abonnement Q du second serveur vers le premier serveur : si un changement effectué sur une colonne non clé dans la table source entre en conflit avec un changement effectué sur la colonne correspondante de la table cible, le programme Q Apply applique dans tous les cas le changement de la table source à la table cible. Le programme Q Apply supprime la ligne en conflit dans la table cible et insère la ligne à partir de la table source. Si une suppression est en conflit et que la ligne est introuvable dans la table cible, le programme Q Apply ignore la suppression à partir de la table source.</p>
	<p>Aucun serveur n'a la priorité.</p>	<p>Le programme Q Apply enregistre tous les conflits dans la table IBMQREP_EXCEPTIONS et poursuit le traitement. Avec le temps, les deux copies d'une table logique différeront.</p>

Tableau 10. Résultats des différentes combinaisons d'options pour la détection de conflit (suite)

Comment contrôler la présence de conflits ?	Comment résoudre les conflits ?	Résultat
Contrôlez uniquement la présence de conflits dans les colonnes non clés.	Le premier serveur est prioritaire.	<p>Pour l'abonnement Q du premier serveur vers le second serveur : si un changement effectué sur une clé de la table source entre en conflit avec la clé de la table cible, le programme Q Apply applique le changement de la table source à la table cible. Le programme Q Apply supprime la ligne en conflit dans la table cible et insère la ligne à partir de la table source. Si une suppression est en conflit et que la ligne est introuvable dans la table cible, le programme Q Apply ignore la suppression à partir de la table source.</p> <p>Pour l'abonnement Q du second serveur vers le premier serveur : si un changement effectué sur une clé de la table source entre en conflit avec la clé de la table cible, le programme Q Apply enregistre le conflit mais ne force pas l'application du changement dans la table cible.</p>
	Le second serveur est prioritaire.	<p>Pour l'abonnement Q du premier serveur vers le second serveur : si un changement effectué sur une clé de la table source entre en conflit avec la clé de la table cible, le programme Q Apply enregistre le conflit mais ne force pas l'application du changement dans la table cible.</p> <p>Pour l'abonnement Q du second serveur vers le premier serveur : si un changement effectué sur une clé de la table source entre en conflit avec la clé de la table cible, le programme Q Apply applique le changement de la table source à la table cible. Le programme Q Apply supprime la ligne en conflit dans la table cible et insère la ligne à partir de la table source. Si une suppression est en conflit et que la ligne est introuvable dans la table cible, le programme Q Apply ignore la suppression à partir de la table source.</p>
	Aucun serveur n'a la priorité.	Le programme Q Apply enregistre tous les conflits dans la table IBMQREP_EXCEPTIONS et poursuit le traitement. Avec le temps, les deux copies d'une table logique différeront.

Création d'abonnements Q pour une réplication entre homologues

Vous pouvez créer des abonnements Q pour mapper des tables entre elles de sorte que les modifications sont répliquées, dans un sens et dans l'autre, depuis chaque table. Cette tâche fait partie de la tâche plus grande concernant la configuration de la réplication de sources vers des cibles (réplication multidirectionnelle).

Création d'abonnements Q pour une réplication entre homologues avec deux serveurs

Dans une réplication entre homologues avec deux serveurs, les changements effectués sur une copie d'une table sont répliqués vers une deuxième copie de cette


même table et les changements de cette deuxième copie sont répliqués en retour vers la première copie. Dans une réplification entre homologues, les lignes et les colonnes sont toutes répliquées et les noms de colonnes dans chaque copie de la table doivent correspondre.

Avant de commencer

Avant de créer des abonnements Q pour une réplification entre homologues, vous devez effectuer les tâches suivantes :

- Créez les tables de contrôle pour les programmes Q Capture et Q Apply sur chaque serveur qui sera utilisé pour une réplification entre homologues. Les tables de contrôle des programmes qui sont sur chaque serveur doivent comporter le même schéma.
- Créez les deux mappes de files de réplification qui transporteront les données entre les deux serveurs. Il vous faut une mappe de files de réplification pour répliquer les données de la première copie de la table vers la seconde et une mappe de files de réplification pour répliquer en retour les données de la deuxième copie vers la première. (Vous pouvez effectuer cette tâche avant ou pendant la création des abonnements Q.)

Restrictions

- Les procédures stockées ne peuvent pas entrer dans une réplification entre homologues.
-  Si la table source comprend un type de colonne LONG VARCHAR, cette table ne peut pas entrer dans une réplification entre homologues. Pour permettre la réplification entre homologues, le Centre de réplification ou l'ASNCLP doit ajouter deux colonnes à la table source. DB2 for z/OS ne permet pas aux outils d'administration de la réplification d'ajouter des colonnes à une table comprenant une colonne LONG VARCHAR.
- Pour les configurations entre homologues et bidirectionnelles, n'utilisez pas l'utilitaire d'importation. Ce dernier consigne les insertions et, par conséquent, celles-ci sont de nouveau capturées.
- Dans une réplification entre homologues et bidirectionnelle, vous devez utiliser les mêmes contraintes dans la source et la cible.
- Les colonnes IDENTITY des tables se trouvant dans les configurations entre homologues doivent être définies comme GENERATED BY DEFAULT.
- La réplification entre homologues n'est pas prise en charge sur les systèmes utilisant IBM HourGlass pour modifier la date et l'heure renvoyées lors d'une demande d'heure. Ce logiciel modifie les colonnes de version requise pour la réplification entre homologues.

A propos de cette tâche

Un abonnement Q est créé de la première table homologue vers la seconde et un autre abonnement est créé de cette seconde table vers la première. Lorsque vous créez des abonnements Q pour une réplification bidirectionnelle à l'aide du programme de ligne de commande ASNCLP ou du Centre de réplification, l'outil d'administration crée ces deux abonnements en une seule fois.

Procédure

Pour créer des abonnements Q pour une réplification entre homologues avec deux serveurs, appliquez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande CREATE QSUB pour la réplification entre homologues. Par exemple, les commandes suivantes définissent l'environnement puis créent deux abonnements Q entre homologues pour la table EMPLOYEE sur les serveurs SAMPLE et SAMPLPEER :</p> <pre>SET SUBGROUP "p2p2group"; SET PEER NODE 1 SERVER DBALIAS SAMPLE SCHEMA GREEN; SET PEER NODE 2 SERVER DBALIAS SAMPLPEER SCHEMA MAGENTA; SET CONNECTION SOURCE SAMPLE.GREEN TARGET SAMPLPEER.MAGENTA REPLQMAP "SAMPLE_GREEN_TO_SAMPLPEER_MAGENTA"; SET CONNECTION SOURCE SAMPLPEER.MAGENTA TARGET SAMPLE.GREEN REPLQMAP "SAMPLPEER_MAGENTA_TO_SAMPLE_GREEN"; SET TABLES (SAMPLE.GREEN.GREEN.EMPLOYEE); CREATE QSUB SUBTYPE P;</pre> <p>Les commandes SET CONNECTION spécifient les deux mappes de files de réplification qui sont utilisées. La commande SET TABLES spécifie la table EMPLOYEE sur le serveur SAMPLE. Une copie correspondante de la table sera créée sur le serveur SAMPLPEER.</p>
Centre de réplification	<p>Utilisez l'assistant Créer des abonnements Q. Pour ouvrir l'assistant, développez le schéma Q Capture ou Q Apply approprié, cliquez avec le bouton droit de la souris sur le dossier Abonnements Q puis sélectionnez Créer.</p> <p>Page Tables cible</p> <p>Examinez le profil objet cible. Modifiez ce profil si nécessaire afin que les tables cible pour les abonnements Q répondent à vos besoins.</p> <p>Le profil objet cible détermine si une table cible existante est utilisée ou si une nouvelle est créée. Le Centre de réplification recherche un objet qui corresponde au système de dénomination défini dans le profil. S'il n'existe aucun objet correspondant, il en est créé un.</p>

Création d'abonnements Q pour une réplification entre homologues avec trois serveurs ou plus

Dans une réplification entre homologues avec trois serveurs ou plus, les changements effectués sur chaque copie d'une table sont répliqués vers toutes les autres copies de cette table. Les lignes et les colonnes sont toutes répliquées et les noms de colonnes dans chaque copie de la table doivent correspondre.

Avant de commencer

Avant de créer des abonnements Q pour une réplification entre homologues, vous devez effectuer les tâches suivantes :

- Créez les tables de contrôle pour les programmes Q Capture et Q Apply sur chaque serveur qui sera utilisé pour une réplication entre homologues. Les tables de contrôle des programmes qui sont sur chaque serveur doivent comporter le même schéma.
- Créez les deux mappes de files de réplication qui transporteront les données entre chaque serveur. Vous devez créer une mappe de files de réplication pour chaque paire source-cible. Si vous souhaitez répliquer une table entre trois serveurs, il vous faut six mappes de files de réplication. Si vous souhaitez répliquer une table entre quatre serveurs, il vous faut douze mappes de files de réplication. (Vous pouvez effectuer cette tâche avant ou pendant la création des abonnements Q.)

Restrictions

- Les vues et les procédures stockées ne peuvent pas entrer dans une réplication entre homologues.
- **z/OS** Si la table source comprend un type de colonne LONG VARCHAR, cette table ne peut pas entrer dans une réplication entre homologues. Pour permettre la réplication entre homologues, le Centre de réplication ou l'ASNCLP doit ajouter deux colonnes à la table source. DB2 for z/OS ne permet pas aux outils d'administration de la réplication d'ajouter des colonnes à une table comprenant une colonne LONG VARCHAR.
- Pour les configurations entre homologues et bidirectionnelles, n'utilisez pas l'utilitaire d'importation. Ce dernier consigne les insertions et, par conséquent, celles-ci sont de nouveau capturées.
- Dans une réplication entre homologues et bidirectionnelle, vous devez utiliser les mêmes contraintes dans la source et la cible.
- Les colonnes IDENTITY des tables se trouvant dans les configurations entre homologues doivent être définies comme GENERATED BY DEFAULT.
- La réplication entre homologues n'est pas prise en charge sur les systèmes utilisant IBM HourGlass pour modifier la date et l'heure renvoyées lors d'une demande d'heure. Ce logiciel modifie les colonnes de version requise pour la réplication entre homologues.

A propos de cette tâche

Un abonnement Q est créé pour chaque couple source vers cible. Si vous souhaitez répliquer une table vers et depuis trois serveurs, six abonnements Q sont créés. Si vous souhaitez répliquer une table vers et depuis quatre serveurs, douze abonnements Q sont créés. La formule $n*(n-1)$, où n représente le nombre de serveurs, sert à déterminer le nombre d'abonnements Q qui sont créés. Lorsque vous créez des abonnements Q pour une réplication entre homologues à l'aide du programme de ligne de commande ASNCLP ou du Centre de réplication, l'outil d'administration crée tous les abonnements nécessaires en une seule fois.

Procédure

Pour créer des abonnements Q pour une réplication entre homologues avec trois serveurs ou plus, appliquez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande CREATE QSUB pour la réplication entre homologues. Par exemple, les commandes suivantes définissent l'environnement puis créent six abonnements Q entre homologues pour la table EMPLOYEE sur les serveurs SAMPLE, SAMPLE2 et SAMPLE3 :</p> <pre> SET SUBGROUP "p2p3group"; SET PEER NODE 1 SERVER DBALIAS SAMPLE SCHEMA GRAY; SET PEER NODE 2 SERVER DBALIAS SAMPLE2 SCHEMA BROWN; SET PEER NODE 3 SERVER DBALIAS SAMPLE3 SCHEMA YELLOW; SET CONNECTION SOURCE SAMPLE.GRAY TARGET SAMPLE2.BROWN REPLQMAP "SAMPLE_GRAY_TO_SAMPLE2_BROWN"; SET CONNECTION SOURCE SAMPLE.GRAY TARGET SAMPLE3.YELLOW REPLQMAP "SAMPLE_GRAY_TO_SAMPLE3_YELLOW"; SET CONNECTION SOURCE SAMPLE2.BROWN TARGET SAMPLE.GRAY REPLQMAP "SAMPLE2_BROWN_TO_SAMPLE_GRAY"; SET CONNECTION SOURCE SAMPLE2.BROWN TARGET SAMPLE3.YELLOW REPLQMAP "SAMPLE2_BROWN_TO_SAMPLE3_YELLOW"; SET CONNECTION SOURCE SAMPLE3.YELLOW TARGET SAMPLE.GRAY REPLQMAP "SAMPLE3_YELLOW_TO_SAMPLE_GRAY"; SET CONNECTION SOURCE SAMPLE3.YELLOW TARGET SAMPLE2.BROWN REPLQMAP "SAMPLE3_YELLOW_TO_SAMPLE2_BROWN"; SET TABLES (SAMPLE.GRAY.GRAY.STAFF); CREATE QSUB SUBTYPE P; </pre> <p>Les commandes SET CONNECTION spécifient les six mappes de files de réplication qui sont utilisés. La commande SET TABLES spécifie la table EMPLOYEE sur le serveur SAMPLE. Des copies correspondantes de la table seront créées sur les serveurs SAMPLE2 et SAMPLE3.</p>

Méthode	Description
Centre de réplication	<p>Utilisez l'assistant Créer des abonnements Q. Pour ouvrir l'assistant, développez le schéma Q Capture ou Q Apply approprié, cliquez avec le bouton droit de la souris sur le dossier Abonnements Q puis sélectionnez Créer.</p> <p>Page Serveurs Lorsque vous spécifiez qu'un serveur servira pour une réplication entre homologues, vous devez spécifier à chaque fois la mappe de files de réplication de ce nouveau serveur vers tous les autres serveurs que vous avez déjà sélectionnés. Vous devez également spécifier la mappe de files de réplication en retour de chaque serveur existant vers le nouveau serveur.</p> <p>Page Tables cible Examinez le profil objet cible pour chaque serveur à l'exception de celui contenant les tables existantes (de base). Modifiez ce profil si nécessaire afin que les tables cible pour les abonnements Q répondent à vos besoins. Le profil objet cible détermine si une table cible existante est utilisée ou si une nouvelle est créée. Le Centre de réplication recherche un objet qui corresponde au système de dénomination défini dans le profil. S'il n'existe aucun objet correspondant, il en est créé un.</p> <p>Page Chargement des tables cible Lorsque vous créez des abonnements Q pour une réplication entre homologues avec trois serveurs ou plus, les abonnements Q sont toujours créés dans un état inactif. C'est à vous de les activer.</p>

Démarrage d'une réplication bidirectionnelle ou d'une réplication entre homologues avec deux serveurs

Pour démarrer une réplication bidirectionnelle ou une réplication entre homologues avec deux serveurs, démarrez les programmes Q Capture et Q Apply sur les deux serveurs puis assurez-vous que les abonnements Q sont activés.

Avant de commencer

Linux UNIX Windows Si une table intervenant dans un abonnement Q entre homologues contient des données avant la création de l'abonnement Q, vous devez exécuter la commande REORG sur la table après avoir créé l'abonnement Q et avant de démarrer les programmes Q Capture sur les serveurs homologues.

A propos de cette tâche

Lorsque vous créez des abonnements Q pour une réplication bidirectionnelle ou la réplication entre homologues avec deux serveurs, les nouveaux abonnements sont automatiquement démarrés par défaut lorsque vous démarrez ou réinitialisez le programme Q Capture. Toutefois, les abonnements Q ne démarrent automatiquement que lorsqu'ils s'agit de nouveaux abonnements. Si vous arrêtez la réplication d'une table logique et que vous voulez redémarrer la réplication, vous devez procéder comme suit pour démarrer la réplication manuellement.

Remarque importante relative aux modifications de la version 9.7 : Le protocole d'initialisation de la réplication multidirectionnelle a changé avec la version 9.7 et la réplication ne se met pas en pause lors de l'activation de tous les abonnements Q. En raison de ce changement, si l'un des programmes Q Capture ou Apply de la configuration est migré vers la version 9.7 et que vous devez ajouter un nouvel abonnement Q ou activer un abonnement Q désactivé, tous les programmes Q Capture et Q Apply de la configuration doivent être à la version 9.7. Si un programme Q Capture impliqué à la fois dans les configurations unidirectionnelles et bidirectionnelles ou entre homologues et que l'un des serveurs est migré vers la version 9.7, tous les composants impliqués dans les configurations unidirectionnelles et bidirectionnelles doivent être migrés vers la version 9.7.

Procédure

Démarrer la réplication d'une table logique dans une réplication bidirectionnelle ou réplication entre homologues avec deux serveurs.

1. Démarrez les programmes Q capture et Q apply sur les deux serveurs. Vous pouvez démarrer les programmes Q Capture à chaud ou à froid :

démarrage à froid

Si vous lancez un démarrage à froid, vous devez démarrer les programmes Q Capture sur chaque serveur avant de démarrer les programmes Q Apply. Vérifiez que Q capture est démarré sur chaque serveur avant de démarrer Q Apply à l'aide de la fenêtre Vérifier le statut du Centre de réplication ou en vérifiant s'il y a un message d'initialisation dans le journal de Q capture.

démarrage à chaud

Si vous lancez un démarrage à chaud, vous pouvez démarrer les programmes Q Apply et Q Capture sur chaque serveur dans l'ordre que vous souhaitez.

Si vous avez créé les abonnements Q de sorte qu'ils soient démarrés automatiquement, la réplication démarre lorsque vous démarrez Q Apply et Q capture.

2. Si vous les avez créés sans démarrage automatique ou si vous redémarrez la réplication, démarrez l'un des deux abonnements pour la table logique. Q Capture et Q Apply démarrent alors automatiquement l'autre abonnement.

Si les abonnements spécifient qu'il y a une phase de chargement, la table source pour l'abonnement que vous démarrez sert à charger la table cible sur l'autre serveur.

Démarrage d'une réplication entre homologues avec trois serveurs ou plus

Après avoir créé les abonnements Q pour une table logique dans une réplication entre homologues, vous devez démarrer le groupe d'abonnements pour lancer la réplication.

Avant de commencer

- Les programmes Q Capture et Q Apply doivent fonctionner sur tous les serveurs du groupe. Vous pouvez démarrer les programmes Q Capture à chaud ou à froid :

démarrage à froid

Si vous lancez un démarrage à froid, vous devez démarrer les programmes Q Capture sur chaque serveur avant de démarrer les programmes Q Apply. Vérifiez que Q capture est démarré sur chaque serveur avant de démarrer Q Apply à l'aide de la fenêtre Vérifier le statut du Centre de réplication ou en vérifiant s'il y a un message d'initialisation dans le journal de Q capture.

démarrage à chaud

Si vous lancez un démarrage à chaud, vous pouvez démarrer les programmes Q Apply et Q Capture sur chaque serveur dans l'ordre que vous souhaitez.

- **Linux UNIX Windows** Si une table intervenant dans un abonnement Q entre homologues contient des données avant la création de l'abonnement Q, vous devez exécuter la commande REORG sur la table après avoir créé l'abonnement Q et avant de démarrer les programmes Q Capture sur les serveurs homologues.

A propos de cette tâche

Vous devez également démarrer le groupe si vous avez arrêté tous les abonnements Q pour une table logique et que vous souhaitez relancer la réplication.

Dans un groupe d'homologues comptant trois serveurs ou plus, le lancement de la réplication s'effectue par étapes. Vous démarrez tout d'abord les abonnements Q pour une table logique entre deux serveurs puis vous ajoutez les nouveaux serveurs un à un, jusqu'à ce qu'ils répliquent tous de manière active cette table logique.

Vous ne pouvez ajouter dans le groupe qu'un seul serveur à la fois. N'ajoutez un nouveau serveur qu'une fois que les autres serveurs répliquent de manière active la table logique.

Remarque importante relative aux modifications de la version 9.7 : Le protocole d'initialisation de la réplication multidirectionnelle a changé avec la version 9.7 et la réplication ne se met pas en pause lors de l'activation de tous les abonnements Q. En raison de ce changement, si l'un des programmes Q Capture ou Apply de la configuration est migré vers la version 9.7 et que vous devez ajouter un nouvel abonnement Q ou activer un abonnement Q désactivé, tous les programmes Q Capture et Q Apply de la configuration doivent être à la version 9.7. Si un programme Q Capture impliqué à la fois dans les configurations unidirectionnelles et bidirectionnelles ou entre homologues et que l'un des serveurs est migré vers la version 9.7, tous les composants impliqués dans les configurations unidirectionnelles et bidirectionnelles doivent être migrés vers la version 9.7.

Procédure

Démarrer une réplication dans un groupe d'homologues comportant trois serveurs ou plus :

1. Choisissez deux serveurs du groupe pour commencer le processus d'activation.
2. Démarrez l'un des deux abonnements Q pour une table logique entre les deux serveurs.

Les programmes Q Capture et Q Apply démarrent automatiquement l'autre abonnement Q pour cette table logique entre les deux serveurs. Si les abonnements spécifient qu'il y a une phase de chargement, l'abonnement que

vous démarrez doit être celui avec la table cible à partir de laquelle vous souhaitez lancer le chargement. Cette table sert à charger la table sur l'autre serveur.

3. Une fois les deux abonnements actifs, choisissez un nouveau serveur à ajouter au groupe.
4. Choisissez l'un des serveurs répliquant activement la table logique.
5. Lancez l'abonnement Q pour la table logique qui spécifie le serveur que vous avez choisi à l'étape 4 comme étant sa source et le nouveau serveur comme étant sa cible.

Les programmes Q Capture et Q Apply démarrent l'autre abonnement Q pour la table logique entre le nouveau serveur et le serveur en cours de réplication active. Les abonnements pour la table logique entre ces serveurs sont également démarrés.

A ce stade, la réplication commence, quel que soit le sens, entre tous les serveurs.

6. Suivez les étapes 3, 4 et 5 jusqu'à ce que tous les abonnements Q du groupe soient actifs.

Arrêt d'une réplication bidirectionnelle ou d'une réplication entre homologues avec deux serveurs

Dans la réplication bidirectionnelle ou la réplication entre homologues avec deux serveurs, vous pouvez arrêter la réplication d'une table logique sans arrêter les programmes Q Capture ou Q Apply. Pour cela, désactivez les abonnements Q pour la table logique. La réplication d'autres tables logiques continuera entre les deux serveurs.

Avant de commencer

Les abonnements Q pour la table logique doivent être à l'état A (actif).

Restrictions

Vous ne pouvez pas arrêter uniquement l'un des deux abonnements Q pour une table logique. Lorsque vous en arrêtez un, l'autre est automatiquement arrêté.

Procédure

Pour arrêter la réplication d'une table logique dans une réplication bidirectionnelle ou dans une réplication entre homologues avec deux serveurs, arrêtez l'un des deux abonnements.

Arrêt d'une réplication entre homologues avec trois serveurs ou plus

Dans la réplication entre homologues avec trois serveurs ou plus, vous pouvez arrêter la réplication d'une table logique sans arrêter les programmes Q Capture ou Q Apply.

Avant de commencer

Les abonnements Q pour la table logique doivent être à l'état A (actif).

A propos de cette tâche

Vous pouvez arrêter la réplication de la table logique aussi bien sur un serveur que sur tous les serveurs du groupe.

Pour arrêter la réplication d'une table logique sur tous les serveurs d'un groupe, suivez la même procédure en traitant les serveurs un par un, jusqu'à ce que tous les abonnements Q pour la table logique soient arrêtés.

Procédure

Pour arrêter la réplication d'une table logique :

1. Choisissez un serveur du groupe répliquant activement la table logique.
2. Arrêtez l'abonnement Q qui spécifie ce serveur comme étant sa source.

Chapitre 7. Réplication des opérations de langage de définition de données (DDL)

Vous pouvez accélérer et simplifier la configuration et la maintenance de solutions de disponibilité continue de DB2 et de reprise après incident grâce aux fonctions de réplication de langage de définition de données (DDL) d'InfoSphere Data Replication version 10.1.3 et supérieure.

Une réplication automatique des opérations DDL (langage de définition de données) élimine ou réduit grandement les besoins en intervention de l'administrateur lors de l'ajout ou du changement de tables au niveau de la base de données principale. Grâce à l'utilisation de cette fonction, les programmes de réplication s'abonnent automatiquement aux tables venant d'être créées pour la réplication et créent ou suppriment ces tables au niveau des cibles.

Une réplication DDL permet de s'assurer que la base de données de reprise reflète vraiment la base de données de secours. Les rubriques suivantes offrent davantage d'informations sur la prise en charge de la réplication DDL.

Abonnements de niveau schéma et réplication DDL

Le mécanisme consistant à répliquer des changements de langage de définition de données (DDL) comme CREATE TABLE ou DROP TABLE entre des bases de données, est un *abonnement de niveau schéma*.

Un abonnement de niveau schéma est un modèle basé sur "*nom_schéma.nom_objet*" qui indique quelles instructions de niveau schéma sont répliquées. Il dit aux programmes de réplication de répondre aux changements DDL selon certains schémas de base de données, et certaines tables au sein de ces schémas, par la capture d'instructions DDL, leur publication sous forme de messages WebSphere et la ré-exécution du DDL au niveau de la base de données cible. Un abonnement de niveau schéma diffère d'un *abonnement Q de niveau table* qui est associé au déplacement de données d'une source spécifique à une cible spécifique.

Chaque partie d'un modèle peut contenir un suffixe avec caractère générique représenté par le signe pourcentage (%).

Un autre composant important des abonnements de niveau schéma est un profil qui précise le mode de création des abonnements Q de niveau table. Le profil permet au programme de ligne de commande ASNCLP et aux programmes de réplication de créer des abonnements Q pour des tables qui correspondent au modèle. Vous créez un abonnement de niveau schéma et le profil avec le programme ASNCLP.

La commande CREATE SCHEMASUB exécute les tâches suivantes :

- Création d'un abonnement de niveau schéma qui inclut les schémas que vous avez indiqués.
- Création d'abonnements Q de niveau table pour toutes les tables au sein des schémas.
- Enregistrement du modèle de schéma et du profil de sorte que les programmes de réplication créent automatiquement des abonnements Q de niveau table avec les options que vous avez spécifiées, pour toutes les tables ajoutées au schéma.

Les informations relatives à l'abonnement de niveau schéma sont enregistrées dans la table IBMQREP_SCHEMASUBS.

Les sections suivantes permettent de voir plus en détail les abonnements de niveau schéma et la réplication DDL :

- «Utilisation des caractères génériques pour indiquer les schémas et tables inclus»
- «Opérations DDL prises en charge pour une réplication», à la page 127
- «Relation entre des abonnements de niveau schéma et des mappes de files d'attente», à la page 127
- «Profils pour la création d'abonnements Q de niveau table», à la page 128
- «Création de tables cible», à la page 128

Utilisation des caractères génériques pour indiquer les schémas et tables inclus

Un abonnement de niveau schéma peut inclure un ou plusieurs schémas, et une ou plusieurs tables au sein de ces schémas. Pour indiquer plusieurs schémas et tables, utilisez le caractère générique pourcentage (%). Ce dernier doit être utilisé comme un suffixe ; par exemple pour un abonnement de niveau schéma "ANU%.%", toute table créée avec un nom de schéma débutant par la chaîne "ANU" est automatiquement répliquée.

Les caractères génériques simplifient grandement la tâche consistant à abonner l'intégralité d'une base de données ; vous pouvez aussi utiliser des caractères génériques pour exclure certains schémas ou certaines tables de l'abonnement de niveau schéma.

Réplication d'une base de données complète

Pour répliquer toutes les opérations CREATE TABLE et DROP TABLE pour toutes les tables de tous les schémas, vous pouvez procéder en créant un abonnement de niveau schéma pour le modèle d'objet schéma : "%.%". Pour cela, dans la commande CREATE SCHEMASUB du programme ASNCLP, il suffit d'utiliser le mot clé ALL, comme dans l'exemple suivant :

```
CREATE SCHEMASUB SUBTYPE B FOR TABLES NODE 1 ALL OPTIONS options1;
```

Cette commande dit : "Créer deux abonnements Q de niveau schéma, un dans chaque sens, pour une configuration de réplication bidirectionnelle ; répliquer CREATE TABLE et DROP TABLE pour toutes les tables de tous les schémas, puis utiliser le profil options1 pour créer les abonnements Q de niveau table nécessaire."

Exclusion de schémas et de tables d'une réplication

Lorsque vous indiquez des noms de schémas et des objets à exclure de l'abonnement de niveau schéma, les exclusions sont enregistrées dans la table de contrôle IBMQREP_EXCLSCHEMA. Par exemple, supposons un abonnement de niveau schéma pour "%.%", sauf qu'il existe une entrée dans la table IBMQREP_EXCLSCHEMA pour "BOB," l'instruction CREATE TABLE BOB.T1 ne doit pas être répliquée, même avec abonnement de niveau schéma actif pour "%.%".

Table IBMQREP_EXCLSCHEMA : indique des schémas et des tables à exclure d'une réplication.

NOM_SCHEMA	NOM_OBJET	QMAP
------------	-----------	------

BOB	T%	SITEA2B
-----	----	---------

Les caractères génériques ne sont pas autorisés lorsque vous excluez des schémas, mais vous pouvez les utiliser pour exclure des tables. L'exclusion ne concerne que l'abonnement de niveau schéma qui utilise une mappe de files de réplication spécifique.

Par défaut, DB2 et les tables du catalogue de réplication sont déjà exclus même si vous utilisez un modèle du type "%.%".

Opérations DDL prises en charge pour une réplication

Pour la première édition des abonnements de niveau schéma, les opérations CREATE TABLE et DROP TABLE sont prises en charge.

Les opérations ALTER TABLE ADD COLUMN et ALTER TABLE ALTER COLUMN SET DATA TYPE peuvent être automatiquement répliquées avec une réplication version 10.1 sous z/OS et Linux, UNIX et Windows. Ces opérations sont exécutées au niveau de la table et sont utilisées pour des colonnes nouvellement ajoutées ; les opérations ne font pas partie de l'abonnement de niveau schéma.

Pour ajouter des colonnes de table source existantes à des abonnements Q, vous pouvez utiliser le tableau de bord de réplication Q, le Centre de réplication ou le programme ASNCLP. Les programmes de réplication ajoutent des colonnes à l'abonnement Q et peuvent également ajouter les colonnes à la table cible. Pour plus d'informations, voir «Ajout de colonnes existantes à un abonnement Q (réplication unidirectionnelle)», à la page 141 et «Ajout de colonnes existantes à un abonnement Q (réplication bidirectionnelle ou entre homologues)», à la page 144.

Vous pouvez aussi indiquer que des opérations LOAD au niveau des tables source sont répliquées. Pour plus d'informations, reportez-vous à la section «Réplication d'opérations de chargement de la table source», à la page 203.

RENAME TABLE n'est pas pris en charge. Si vous renommez une table, vous devez arrêter l'abonnement Q de niveau table, mettre à jour les propriétés de l'abonnement Q avec le nouveau nom de table et réinitialiser le programme Q Capture.

Relation entre des abonnements de niveau schéma et des mappes de files d'attente

Vous ne pouvez pas avoir plusieurs abonnements Q de niveau table pour la même table source qui indique la même mappe de files de réplication, et des modèles d'abonnements de niveau schéma associés à la même mappe de files d'attente ne doivent pas se chevaucher. Par exemple, les modèles "SERGE.%" et "%.T1" correspondent tous les deux à la table SERGE.T1. Ce type de chevauchement n'est pas autorisé et le programme ASNCLP impose la restriction lors de la création des abonnements de niveau schéma.

Par exemple, si un abonnement de niveau schéma avec le modèle "ANUPAMA.%" existe déjà, vous ne pouvez pas ajouter un abonnement de niveau schéma avec le modèle "%.%", car des tables pourraient correspondre aux deux modèles. La commande CREATE SCHEMASUB devrait échouer.

Vous pouvez avoir plusieurs abonnements de niveau schéma pour le même ensemble de tables, mais pour des mappes de files d'attente différentes. Par exemple :

- %.% dans une réplication bidirectionnelle sur QUEUEMAP1, pour une reprise après incident
- ANUPAMA.% dans une réplication unidirectionnelle sur QUEUEMAP2, pour alimenter un entrepôt de données

Profils pour la création d'abonnements Q de niveau table

Le profil que vous créez avec la commande CREATE SUBSCRIPTION OPTIONS du programme ASNCLP est utilisé pour deux raisons :

- Le programme ASNCLP crée des abonnements Q de niveau table pour des tables existantes qui correspondent au modèle d'abonnement de niveau schéma.
- Les programmes de réplication créent des abonnements Q de niveau table en réponse à une opération CREATE TABLE qui correspond au modèle.

La création du profil est l'une des premières choses à faire lors de la configuration d'une réplication DDL, car vous devez indiquer le profil dans la commande CREATE SCHEMASUB. Si vous ne créez pas de profil, le programme ASNCLP et les programmes de réplication utilisent des profils par défaut pour des réplications unidirectionnelles et bidirectionnelles.

Les profils sont enregistrés dans la table IBMQREP_SUBS_PROF du serveur Q Capture.

Pour plus d'informations, voir «Création de profils pour des abonnements Q de niveau table», à la page 134.

Création de tables cible

Lorsqu'il crée des tables cible après une opération CREATE TABLE au niveau de la source, le programme Q Apply crée la table cible avec tous les index et contraintes qui ont existé à la source, même si ceux-ci ont été créés avec plusieurs instructions ALTER et dans plusieurs transactions.

Le programme Q Apply réexécute les instructions DDL comme si elles étaient exécutées à la source, sauf qu'il émet une instruction SET CURRENT DEFAULT SCHEMA avant de créer chaque table, puis change son appartenance pour la même que celle de l'utilisateur qui a créé l'objet au niveau de la base de données source (sinon, le créateur de l'objet devrait être l'ID utilisateur qui a lancé le programme Q Apply).

Pour des tables source existantes, le programme ASNCLP utilise les options indiquées dans la commande SET PROFILE pour créer des tables cibles lorsqu'il traite une commande CREATE SCHEMASUB.

Gestion des opérations DDL par Q Capture au niveau de la base de données source

Le programme Q Capture réplique automatiquement certaines opérations DDL au niveau de la base de données source. D'autres changements DDL nécessitent une action de votre part sur la base de données cible.

Le tableau suivant décrit comment Q Capture gère différents types d'opérations DDL et comment agir sur tout abonnement Q affecté.

Tableau 11. Gestion des changements DDL à la base de données source par Q Capture et mode d'action

Opération de langage de définition de données	Comment elle est traitée	Action à accomplir
CREATE TABLE	<p>Version 10.1 sous Linux, UNIX et Windows Répliqué automatiquement si la nouvelle table source correspond au modèle d'attribution des noms des schémas et des tables dans l'abonnement au niveau schéma. Lorsque le programme Q Capture détecte dans le journal une opération CREATE TABLE qui correspond à un abonnement au niveau schéma, il informe le programme Q Apply afin de créer une table cible correspondante. Un abonnement Q au niveau table est également créé et mappe les nouvelles tables source et cible.</p> <p>z/OS ou versions antérieures sous Linux, UNIX et Windows Aucune réplification automatique des opérations CREATE TABLE.</p>	<p>Version 10.1 sous Linux, UNIX et Windows Vérifiez que la table source nouvellement créée correspond au modèle d'attribution des noms des schémas et des tables dans un abonnement au niveau schéma.</p> <p>z/OS ou versions antérieures sous Linux, UNIX et Windows Créez un abonnement Q de niveau table pour la nouvelle table source et utilisez les outils d'administration de réplication pour créer une table cible correspondante ou utiliser une table cible existante.</p>

Tableau 11. Gestion des changements DDL à la base de données source par Q Capture et mode d'action (suite)

Opération de langage de définition de données	Comment elle est traitée	Action à accomplir
DROP TABLE	<p>Version 10.1 sous Linux, UNIX et Windows Répliqué automatiquement si la table source fait partie d'un souscription de niveau schéma. Lorsque le programme Q Capture détecte dans le journal une opération DROP TABLE qui correspond à un abonnement de niveau schéma, les abonnements Q de niveau table associés pour toutes les files d'attente sont également supprimés.</p> <p>z/OS ou versions antérieures sous Linux, UNIX et Windows Q Capture laisse l'abonnement Q actif mais la table source n'a aucun enregistrement de journal à lire. Sous z/OS, le message d'avertissement ASN0197W est émis.</p>	<p>Version 10.1 sous Linux, UNIX et Windows Vérifiez que la table source est incluse dans un abonnement de niveau schéma.</p> <p>z/OS ou versions antérieures sous Linux, UNIX et Windows Lorsque vous supprimez une table, l'abonnement Q pour la table existe toujours. Pour la suppression, arrêtez l'abonnement Q, puis supprimez-le.</p>

Tableau 11. Gestion des changements DDL à la base de données source par Q Capture et mode d'action (suite)

Opération de langage de définition de données	Comment elle est traitée	Action à accomplir
<p>ALTER TABLE ADD (COLUMN)</p>	<p>Version 10.1 sous z/OS et Linux, UNIX ou Windows ou ultérieure Si vous définissez la valeur de la colonne REPL_ADDCOL à Y dans la table IBMQREP_SUBS, lorsque vous ajoutez de nouvelles colonnes à une table, les colonnes sont ajoutées automatiquement à l'abonnement Q, puis à la table cible si elles n'existent pas déjà dans celle-ci.</p> <p>Versions antérieures Q Capture laisse l'abonnement Q actif mais ne réplique pas la colonne ajoutée jusqu'à la réception d'un signal ADDCOL.</p>	<p>Version 10.1 sous z/OS et Linux, UNIX ou Windows ou ultérieure Indiquez REPLICATE ADD COLUMN YES dans la commande ASNCLP CREATE QSUB ou activez la case à cocher Automatically replicate new columns added to the source table (Répliquer automatiquement de nouvelles colonnes ajoutées à la table source) lorsque vous créez ou modifiez les propriétés d'un abonnement Q dans le Centre de réplication. Pour plus d'informations, consultez ces rubriques :</p> <ul style="list-style-type: none"> • z/OS Enabling replication of ADD COLUMN and SET DATA TYPE operations (Activation de la réplication des opérations ADD COLUMN et SET DATA TYPE) • «Activation d'une réplication automatique des colonnes nouvellement ajoutées à partir de la table source», à la page 140 <p>Versions antérieures Utilisez le Tableau de bord de réplication Q ou la commande ASNCLP ALTER ADD COLUMN ou insérez manuellement un signal ADDCOL pour indiquer que vous voulez répliquer la nouvelle colonne.</p>
<p>TRUNCATE TABLE</p>	<p>z/OS Une opération TRUNCATE est consignée similairement à une suppression globale, de sorte que l'opération est répliquée en tant que série de suppressions à lignes uniques.</p> <p>Linux UNIX Windows La réplication des opérations TRUNCATE n'est pas prise en charge.</p>	<p>z/OS Aucune action n'est requise. Si la table cible possède des lignes qui ne sont pas dans la table source, ces lignes ne sont pas supprimées.</p> <p>Linux UNIX Windows Si vous avez besoin d'effectuer une opération TRUNCATE sur une table cible en plus de sa source, vous devez émettre l'instruction TRUNCATE directement par rapport à la table cible.</p>

Tableau 11. Gestion des changements DDL à la base de données source par Q Capture et mode d'action (suite)

Opération de langage de définition de données	Comment elle est traitée	Action à accomplir
ALTER TABLE ALTER COLUMN SET DATA TYPE	Répliqué automatiquement pour la version 10.1 et ultérieure. Le type de données de la colonne de la table cible correspondante est modifié et la réplication se poursuit normalement.	<p>z/OS</p> <p>Voir Enabling replication of ADD COLUMN and SET DATA TYPE operations (Activation de la réplication des opérations ADD COLUMN et SET DATA TYPE).</p> <p>Linux UNIX Windows</p> <p>Voir «Réplication automatique des opérations ALTER TABLE ALTER COLUMN SET DATA TYPE», à la page 146.</p>
Autre DDL qui modifie la structure d'une table	Q Capture laisse l'abonnement Q inchangé.	<ol style="list-style-type: none"> 1. Arrêtez l'abonnement Q. 2. Modifiez les tables source et cible. 3. Démarrez l'abonnement Q.
Langage de définition de données qui n'altère pas la structure d'une table Par exemple : • CREATE INDEX • ALTER FOREIGN KEY • ADD CONSTRAINT	Q Capture laisse l'abonnement Q actif.	Vérifiez que les contraintes d'unicité, les clés principales et les contraintes d'intégrité référentielle correspondent entre les tables source et cible. Si vous modifiez l'une de ces propriétés sur la table source, effectuez la modification correspondante sur la table cible pour éviter un comportement imprévu. Redémarrez également le programme Q Apply pour qu'il prenne en compte la modification.

Création d'abonnements de niveau schéma

Vous créez un abonnement de niveau schéma pour répliquer des opérations DDL prises en charge comme CREATE TABLE et DROP TABLE pour un ou plusieurs schémas d'une base de données.

Avant de commencer

L'ID utilisateur qui démarre le programme Q Apply doit avoir l'autorisation de créer et de supprimer des tables dans la base de données cible.

A propos de cette tâche

Lorsque vous créez ce type d'abonnement, les programmes de réplication créent automatiquement des abonnements Q de niveau table lorsqu'ils traitent des enregistrements de journaux destinés à des opérations CREATE TABLE qui correspondent au modèle d'attribution de noms de schémas et de tables pour l'abonnement. D'autres opérations DDL prises en charge au sein du schéma sont également répliquées.

Pour une réplication bidirectionnelle, un abonnement au niveau schéma est requis pour chaque sens de réplication. Si vous utilisez le programme de ligne de commande ASNCLP, il crée les ensembles appariés des abonnements de niveau schéma à l'aide d'une commande unique.

La commande ASNCLP peut également créer les abonnements Q de niveau table requis pour des tables existantes qui correspondent au modèle d'attribution de noms des tables.

Restrictions

Les configurations suivantes ne sont pas prises en charge :

- Réplication entre homologues
- Cibles CCD
- Cibles fédérées
- Cibles de procédure stockée
- Réplication entre DB2 for z/OS et DB2 for Linux, UNIX et Windows

Les types de DDL suivants ne sont pas répliqués :

- Opérations ADD, ATTACH et DETACH pour des tables partitionnées
- Séquences de base de données.
- CREATE ou DROP des déclencheurs ou des tables de requêtes matérialisées
- **Linux UNIX Windows** CREATE TABLESPACE

Autres restrictions :

- Les abonnements Q de niveau table qui font partie d'un abonnement de niveau schéma ne peuvent pas utiliser des conditions de recherche Q Capture ni des expressions SQL Q Apply.
- Pour une réplication bidirectionnelle, une table doit être utilisée pour des opérations DML ou LOAD sur un seul site jusqu'à ce que tous les abonnements Q de niveau table créés automatiquement soient créés et actifs sur les deux sites. Par exemple, dans une configuration bidirectionnelle, si vous créez une table sur le site 1 et le site 2, vous pouvez démarrer le chargement de la table sur le site 1, mais ne devez effectuer aucune opération DML ou LOAD sur le site 2 tant que tous les abonnements Q n'ont pas été créés sur le site 2 et ne sont pas dans l'état actif (A).

Remarque : La liste des opérations non prises en charge n'inclut pas toutes les limitations de réplication DDL possibles, mais répertorie des exclusions notables. Si rien n'est répertorié comme étant pris en charge, le prise en charge ne s'effectue pas.

Procédure

Pour créer un abonnement de niveau schéma, utilisez la commande CREATE SCHEMASUB du programme ASNCLP.

La commande crée des abonnements Q de niveau table pour toutes les tables du schéma qui satisfont le modèle d'attribution de noms que vous avez indiqué. Elle enregistre également le modèle de schéma pour que les programmes de réplication créent automatiquement des abonnements Q pour toutes les tables qui ont été ajoutées au sein du schéma.

Vous pouvez créer un abonnement de niveau schéma qui inclut plusieurs schémas en utilisant le signe pourcentage (%) comme caractère générique. Pour répliquer toutes les opérations CREATE TABLE et DROP TABLE dans tous les schémas de la base de données, indiquez le mot clé ALL (qui est équivalent à OWNER LIKE % NAME LIKE % et enregistré comme %.%).

Vous pouvez éventuellement inclure le nom d'un profil pour la création d'abonnement Q de niveau table et permettre aux programmes ASNCLP et de réplication d'utiliser les options que vous indiquez dans le profil.

L'exemple suivant crée un abonnement de niveau schéma appelé sampschemasub1 pour une réplication bidirectionnelle. L'abonnement inclut tous les schémas et tables de la base de données SAMPLE1 et utilise le profil enregistré options1 :

```
SET BIDI NODE 1 SERVER DBALIAS SAMPLE1;
SET BIDI NODE 2 SERVER DBALIAS SAMPLE2;

CREATE SCHEMASUB "sampschemasub1" SUBTYPE B FOR TABLES NODE 1 ALL OPTIONS options1;
```

Création de profils pour des abonnements Q de niveau table

Vous pouvez créer un profil indiquant les options que les programmes de réplication doivent utiliser lorsqu'ils créent automatiquement des abonnements Q de niveau table associés à un abonnement de niveau schéma.

A propos de cette tâche

Le programme ASNCLP utilise également le profil lorsqu'il crée automatiquement des abonnements Q de niveau table pour toutes les tables d'un ou plusieurs schémas spécifiques, en réponse à une commande CREATE SCHEMASUB.

Le profil fournit des valeurs pour l'établissement de l'abonnement Q sur les serveurs Q Capture et Q Apply. Les programmes de réplication utilisent des valeurs issues du profil pour remplir la table IBMQREP_SUBS au niveau du serveur Q Capture et la table IBMQREP_TARGETS au niveau du serveur Q Apply.

Q Capture lit le profil lorsque l'abonnement de niveau schéma est tout d'abord démarré ou lorsque Q Capture est réinitialisé. Le changement d'un profil ne change pas un abonnement Q de niveau table existant. Pour modifier des abonnements Q existants, utilisez les outils d'administration de réplication pour changer les options d'abonnement Q, puis émettez une commande **reinit** pour permettre à Q Capture de lire vos changements.

Si vous ne créez pas de profil, les programmes de réplication utilisent alors des profils par défaut pour la création d'abonnements Q de niveau table. Ces profils diffèrent selon le type d'abonnements Q (unidirectionnel ou bidirectionnel) et sont intitulés ASNBIDI et ASNUNI.

Important : Ne créez pas de profil qui doit se traduire par la création d'abonnement Q de niveau table qui diffèrent des abonnements Q de niveau table existants au sein du même abonnement de niveau schéma. Tous les abonnements de niveau table associés à un abonnement de niveau schéma commun doivent être cohérents.

Procédure

Pour créer un profil pour des abonnements Q de niveau table, utilisez la commande CREATE SUBSCRIPTION OPTIONS du programme de ligne de commande ASNCLP.

Dans la commande, indiquez autant d'options que nécessaires pour la création d'un abonnement Q de niveau table. Par exemple, la commande suivante relative à une réplication bidirectionnelle indique qu'en cas de conflits de lignes, la ligne de la table source est utilisée (CONFLICT ACTION F), que les opérations de chargement au niveau des tables source sont répliquées (CAPTURE LOAD R) et que les opérations de suppression en cascade ne sont pas répliquées (IGNORE CASCADE DELETES) :

```
SET BIDI NODE 1 SERVER SAMPLE;
SET BIDI NODE 2 SERVER SAMPLE2;

CREATE SUBSCRIPTION OPTIONS bidioptions
SUBTYPE B, CONFLICT ACTION F, CAPTURE LOAD W, IGNORE CASCADE DELETES;
```

Si vous indiquez ce profil dans la commande CREATE SCHEMASUB, le programme ASNCLP utilise les options du profil lors de la création des abonnements Q de niveau table pour tous les schémas et tables spécifiés, et les programmes Q Capture et Q Apply utilisent le profil lorsqu'ils créent des abonnements Q pour les tables source venant d'être créées en réponse à une opération CREATE TABLE.

Attribut DATA CAPTURE CHANGES et abonnements de niveau schéma

Pour activer la réplication des opérations CREATE TABLE, l'attribut DATA CAPTURE CHANGES des nouvelles tables qui font partie d'un abonnement de niveau schéma doit être défini.

L'attribut DATA CAPTURE CHANGES garantit que DB2 fournit des informations supplémentaires dans son journal de reprise concernant les changements SQL apportés à la table. Vous pouvez activer l'attribut pour les tables venant d'être créées en utilisant l'une des méthodes suivantes :

- Indiquez explicitement l'attribut DATA CAPTURE CHANGES dans l'instruction CREATE TABLE.
- **Linux UNIX Windows** Utilisez l'attribut DATA CAPTURE CHANGES avec l'instruction CREATE SCHEMA. Avec cette option, toutes les tables créées au sein du schéma spécifié ont l'attribut DATA CAPTURE CHANGES défini par défaut.
- **Linux UNIX Windows** Définissez le paramètre de configuration de niveau base de données `dft_schemas_dcc` à YES. Par défaut, `dft_schemas_dcc` est défini à NO. En le définissant à YES, tous les schémas créés ont par défaut la clause DATA CAPTURE CHANGES.

Si la commande CREATE TABLE indique explicitement DATA CAPTURE NONE, les programmes de réplication considèrent que vous ne voulez pas répliquer cette table et émettent un message d'avertissement pour que vous puissiez modifier l'attribut DATA CAPTURE le cas échéant.

Démarrage d'abonnements de niveau schéma

Vous démarrez un abonnement de niveau schéma pour informer le programme Q Capture qu'il peut commencer à capturer des opérations SQL comme CREATE TABLE et DROP TABLE pour des tables faisant partie de l'abonnement. De la même manière, vous pouvez éventuellement démarrer tous les abonnements Q de niveau table inclus.

A propos de cette tâche

Les abonnements de niveau schéma récemment créés sont à l'état N (nouveau) et sont démarrés automatiquement lorsque le programme Q Capture est démarré ou réinitialisé. Cette procédure pourrait être utile si vous arrêtez l'abonnement ou s'il a été arrêté en raison d'une erreur.

Le démarrage d'un abonnement de niveau schéma implique l'insertion d'un signal START SCHEMASUB dans la table IBMQREP_SIGNAL au niveau d'un ou plusieurs serveurs Q Capture. Lorsque Q Capture traite le signal, l'état de l'abonnement de niveau schéma correspondant change pour A (active) dans la table IBMQREP_SCHEMASUBS.

Lorsque vous utilisez le programme ASNCLP pour démarrer des abonnements de niveau schéma, celui-ci exécute les insertions du signal. Si vous utilisez SQL pour effectuer vous-même les insertions, le signal START SCHEMASUB doit être inséré dans la table IBMQREP_SIGNAL au niveau de tous les serveurs Q Capture dans des configurations bidirectionnelles ou entre homologues.

Procédure

Pour démarrer un abonnement de niveau schéma, utilisez l'une des méthodes suivantes :

Méthode	Description
<p>Programme de ligne de commande ASNCLP</p>	<p>Utilisez la commande START SCHEMASUB. Vous avez deux options :</p> <p>Démarrer l'abonnement de niveau schéma uniquement Cette option informe le programme Q Capture qu'il peut commencer à répliquer des opérations CREATE TABLE et DROP TABLE pour toutes les tables faisant partie de l'abonnement de niveau schéma. L'exemple suivant réalise cette tâche pour schemasub1 entre les serveurs de réplication bidirectionnelle SAMPLE1 et SAMPLE2 :</p> <pre>ASNCLP SESSION SET TO Q REPLICATION; SET BIDI NODE 1 SERVER DBALIAS SAMPLE1; SET BIDI NODE 2 SERVER DBALIAS SAMPLE2; SET RUN SCRIPT NOW STOP ON SQL ERROR ON; START SCHEMASUB schemasub1 NEW ONLY;</pre> <p>Démarrer l'abonnement de niveau schéma et tous les abonnements Q de niveau table qui lui sont associés Cette option informe le programme Q Capture qu'il peut commencer à répliquer des opérations DDL prises en charge pour toutes les tables faisant partie de l'abonnement de niveau schéma et de démarrer la réplication des changements de lignes pour l'ensemble des abonnements Q de niveau table faisant partie de l'abonnement de niveau schéma.</p> <p>Dans l'exemple suivant, les autres commandes qui sont requises dans le script sont les mêmes que celles de l'exemple précédent et ne sont pas reproduites dans l'exemple :</p> <pre>START SCHEMASUB schemasub1 ALL;</pre>

Méthode	Description
SQL	<p>Insérez un signal START SCHEMASUB dans la table IBMQREP_SIGNAL au niveau d'un ou plusieurs serveurs Q Capture :</p> <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'START SCHEMASUB', 'sous-nom_schéma', 'P');</pre> <p>Où <i>schéma</i> identifie un programme Q Capture et <i>sous-nom_schéma</i> représente le nom de l'abonnement de niveau schéma que vous voulez démarrer.</p> <p>Pour une réplication bidirectionnelle ou entre homologues, vous devez insérer le signal dans tous les serveurs Q Capture de la configuration.</p>

Changement des options de niveau table utilisés par des abonnements de niveau schéma

Vous ne pouvez pas mettre à jour le profil qui permet de créer des abonnements Q de niveau table, mais vous pouvez créer un profil avec les changements requis, puis pointer sur l'abonnement de niveau schéma pour commencer à utiliser le nouveau profil.

A propos de cette tâche

L'impossibilité de mettre à jour des profils s'explique par le fait que d'autres abonnements de niveau schéma pourraient utiliser le profil que vous souhaitez changer. Pour commencer à utiliser un nouveau profil, utilisez des commandes ASNCLP et une instruction de mise à jour SQL.

Un changement pour un nouveau profil n'affecte que les abonnements Q futurs qui sont créés pour de nouvelles tables. Pour mettre à jour des abonnements Q de niveau table existants, vous devez émettre une autre instruction SQL.

Procédure

1. Créez un profil pour des abonnements Q de niveau table qui contiennent les nouvelles options. Utilisez la commande CREATE SUBSCRIPTION OPTIONS dans la commande ASNCLP.

L'exemple suivant crée un profil appelé *bidilist_new* pour une configuration bidirectionnelle. Le nouveau profil indique une action d'erreur S (Q Apply s'arrête), un changement de l'action d'erreur par défaut Q (Q Apply arrête de lire dans la file d'attente de réception).

```
SET BIDI NODE 1 SERVER SAMPLE1;
SET BIDI NODE 2 SERVER SAMPLE2;
CREATE SUBSCRIPTION OPTIONS bidilist_new
SUBTYPE B
ERROR ACTION S;
```

2. Mettez à jour la table IBMQREP_SCHEMASUBS pour pointer vers la nouvelle liste d'options. Pour une réplification bidirectionnelle, mettez à jour la table IBMQREP_SCHEMASUBS sur les deux serveurs.

En utilisant une configuration bidirectionnelle à titre d'exemple, vous devez émettre l'instruction de mise à jour suivante sur les deux serveurs SAMPLE1 et SAMPLE2 :

```
UPDATE ASN.IBMQREP_SCHEMASUBS SET SUBPROFNAME = 'bidilist_new'  
WHERE SCHEMA_SUBNAME = 'schemasub1';
```

3. Mettez à jour les abonnements Q de niveau table pour changer l'action d'erreur pour S en émettant l'instruction SQL suivante pour les tables IBMQREP_TARGETS sur les deux serveurs.

```
UPDATE ASN.IBMQREP_TARGETS SET ERROR_ACTION = 'S'  
nWHERE SCHEMA_SUBNAME = 'schemasub1';
```

4. Réinitialisez l'abonnement de niveau schéma et l'ensemble de ses abonnements Q de niveau table à l'aide de la commande REINIT SCHEMASUB avec le mot clé ALL dans la commande ASNCLP. Cette commande invite le programme Q Capture à lire les changements que vous avez apportés aux tables de contrôle.

```
SET BIDI NODE 1 SERVER SAMPLE1;  
SET BIDI NODE 2 SERVER SAMPLE2;
```

```
REINIT SCHEMASUB schemasub1 ALL;
```

Arrêt des abonnements de niveau schéma

Vous arrêtez un abonnement de niveau schéma pour informer le programme Q Capture qu'il peut cesser de capturer des opérations SQL comme CREATE TABLE et DROP TABLE pour des tables faisant partie de l'abonnement. De la même manière, vous pouvez éventuellement arrêter tous les abonnements Q de niveau table inclus.

A propos de cette tâche

L'arrêt d'un abonnement de niveau schéma implique l'insertion d'un signal STOP SCHEMASUB dans la table IBMQREP_SIGNAL au niveau d'un ou plusieurs serveurs Q Capture. Lorsque vous utilisez le programme ASNCLP pour arrêter des abonnements de niveau schéma, celui-ci exécute les insertions.

Lorsque Q Capture traite le signal, l'état de l'abonnement de niveau schéma correspondant change pour I (inactif) dans la table IBMQREP_SCHEMASUBS. L'état des abonnements Q de niveau table reste inchangé tant que vous n'avez pas explicitement arrêté ces abonnements, ce qui peut être effectué au moyen d'une unique commande du programme ASNCLP.

Si vous utilisez SQL pour effectuer vous-même les insertions, le signal STOP SCHEMASUB doit être inséré dans la table IBMQREP_SIGNAL au niveau de tous les serveurs Q Capture dans des configurations bidirectionnelles ou entre homologues.

Procédure

Pour arrêter un abonnement de niveau schéma, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande STOP SCHEMASUB. Vous avez deux options :</p> <p>Arrêter l'abonnement de niveau schéma uniquement</p> <p>Cette option informe le programme Q Capture qu'il peut cesser de répliquer des opérations CREATE TABLE et DROP TABLE pour toutes les tables faisant partie de l'abonnement de niveau schéma. L'exemple suivant réalise cette tâche pour schemasub1 entre les serveurs de réplification bidirectionnelle SAMPLE1 et SAMPLE2 :</p> <pre>ASNCLP SESSION SET TO Q REPLICATION;</pre> <pre>SET BIDI NODE 1 SERVER DBALIAS SAMPLE1;</pre> <pre>SET BIDI NODE 2 SERVER DBALIAS SAMPLE2;</pre> <pre>SET RUN SCRIPT NOW STOP ON SQL ERROR ON;</pre> <pre>STOP SCHEMASUB schemasub1 NEW ONLY;</pre> <p>Arrêter l'abonnement de niveau schéma et tous les abonnements Q de niveau table qui lui sont associés</p> <p>Cette option informe le programme Q Capture qu'il peut cesser de répliquer des opérations DDL prises en charge pour toutes les tables faisant partie de l'abonnement de niveau schéma et de mettre fin à la réplification des changements de lignes pour l'ensemble des abonnements Q de niveau table faisant partie de l'abonnement de niveau schéma.</p> <p>Dans l'exemple suivant, les autres commandes qui sont requises dans le script sont les mêmes que celles de l'exemple précédent et ne sont pas reproduites dans l'exemple :</p> <pre>STOP SCHEMASUB schemasub1 ALL;</pre>
SQL	<p>Insérez un signal STOP SCHEMASUB dans la table IBMQREP_SIGNAL au niveau d'un ou plusieurs serveurs Q Capture :</p> <pre>insert into <i>schéma</i>.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'STOP SCHEMASUB', '<i>sous-nom_schéma</i>', 'P');</pre> <p>Où <i>schéma</i> identifie un programme Q Capture et <i>sous-nom_schéma</i> représente le nom de l'abonnement de niveau schéma que vous souhaitez arrêter.</p> <p>Pour une réplification bidirectionnelle ou entre homologues, vous devez insérer le signal dans tous les serveurs Q Capture de la configuration.</p>

Suppression d'abonnements de niveau schéma

Vous pouvez supprimer des abonnements de niveau schéma qui ne sont pas activement traités, et éventuellement indiquer la suppression de tous les abonnements Q de niveau table associés.

A propos de cette tâche

La commande DROP SCHEMASUB du programme ASNCLP vous permet de ne supprimer que l'abonnement de niveau schéma (avec les mots clé NEW ONLY) ou de supprimer l'ensemble des abonnements Q de niveau table (avec le mot clé ALL).

Procédure

1. Utilisez le tableau de bord de réplication Q, le Centre de réplication ou SQL pour vérifier que l'abonnement de niveau schéma est à l'état inactif (I).
2. Utilisez l'une de ces méthodes pour vérifier que tous les abonnements Q de niveau table associés sont inactifs.
3. Emettez la commande DROP SCHEMASUB.

L'exemple suivant supprime l'abonnement de niveau schéma JASLY10 d'une configuration bidirectionnelle et supprime tous les abonnements Q de niveau table qui lui sont associés.

```
SET BIDI NODE 1 SERVER SAMPLE;  
SET BIDI NODE 2 SERVER SAMPLE2;
```


```
DROP SCHEMASUB JASLY10 ALL;
```

Les entrées liées à l'abonnement sont supprimées des tables de contrôle de Q Capture et Q Apply sur les deux serveurs dans la configuration bidirectionnelle.

Activation d'une réplication automatique des colonnes nouvellement ajoutées à partir de la table source

Vous pouvez configurer vos abonnements Q pour que, lorsque de nouvelles colonnes sont ajoutées à la table source, celles-ci soient automatiquement ajoutées à la table cible et répliquées.

Avant de commencer

- Les serveurs Q Capture et Q Apply doivent être à la version 10.1 ou plus récente à la fois sous z/OS et Linux, UNIX et Windows.
-  Certaines étapes de configurations sont nécessaires sous z/OS. Voir Enabling replication of ADD COLUMN and SET DATA TYPE operations (Activation de la réplication des opérations ADD COLUMN et SET DATA TYPE).
- Vérifiez que l'ID utilisateur qui exécute le programme Q Apply dispose du privilège ALTER sur toutes les tables cible auxquelles vous souhaitez ajouter de nouvelles colonnes.

Restrictions

Cette fonction n'est pas prise en charge dans les situations suivantes :

- Vous ajoutez des colonnes existantes à un abonnement Q

- Vous voulez indiquer un nom de colonne avant image pour la table de modification cohérente des données (CCD) cible
- Vous voulez indiquer un nom différent pour la colonne cible

Dans ces cas, utilisez le signal ADDCOL décrit dans «Ajout de colonnes existantes à un abonnement Q (réplication unidirectionnelle)» et «Ajout de colonnes existantes à un abonnement Q (réplication bidirectionnelle ou entre homologues)», à la page 144.

Procédure

Pour activer une réplication automatique des colonnes nouvellement ajoutées, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Dans la commande CREATE QSUB, indiquez REPLICATE ADD COLUMN YES. L'exemple suivant illustre les commandes de configuration, puis la commande CREATE QSUB avec cette option :</p> <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DBALIAS SAMPLE; SET SERVER TARGET TO DBALIAS TARGETDB; SET RUN SCRIPT NOW STOP ON SQL ERROR ON;</pre> <pre>CREATE QSUB USING REPLQMAP SAMPLE ASN TO _TARGETDB_ASN (SUBNAME EMPLOYEE0001 db2admin.EMPLOYEE OPTIONS HAS LOAD PHASE I REPLICATE ADD COLUMN YES KEYS (EMPNO) LOAD TYPE 1);</pre>
Centre de réplication	<p>Activez la case à cocher Automatically replicate new columns added to the source table (Répliquer automatiquement de nouvelles colonnes ajoutées à la table source) dans l'une de ces fenêtres ou sur les pages de l'assistant :</p> <ul style="list-style-type: none"> • Colonnes de table source (abonnement Q unidirectionnel unique) • Quelles colonnes source mappent vers quelle colonnes cible ? (plusieurs abonnements Q unidirectionnels) • Quels paramètres de profil pour les tables cible ? (réplication bidirectionnelle)

Ajout de colonnes existantes à un abonnement Q (réplication unidirectionnelle)

Vous pouvez ajouter des colonnes existantes à un abonnement Q unidirectionnel à partir de la table source lors de l'exécution des programmes de réplication. Si les colonnes n'existent pas et que vos serveurs sont à la version 10.1 ou ultérieure, utilisez une procédure différente.

Avant de commencer


- Les abonnements Q auxquels les colonnes sont ajoutées doivent avoir pour état A (active).
- **Linux UNIX Windows** Si le type de données de la colonne est LONG VARCHAR ou GRAPHIC, la base de données source ou le sous-système doit être configuré avec DATA CAPTURE CHANGES INCLUDE VARCHAR COLUMNS.

Restrictions

- Les colonnes que vous ajoutez doivent avoir pour valeur NULL admis ou être définies comme NOT NULL WITH DEFAULT.
- Si vous ajoutez des colonnes avec des valeurs par défaut, vous devez exécuter l'utilitaire REORG dans la table source avant de démarrer la réplication de la nouvelle colonne. Pour plus d'informations, voir Eviter les erreurs CHECKFAILED lors de l'ajout de colonnes aux tables cible DB2 for z/OS.
- Vous ne pouvez pas ajouter plus de 20 colonnes au cours d'un intervalle de validation Q Capture, tel que spécifié par le paramètre **commit_interval**.
- **Cibles fédérées** : pour ajouter des colonnes à un abonnement Q existant, vous pouvez utiliser le signal ADDCOL mais vous devez déplacer l'abonnement puis le créer de nouveau après avoir modifié la table cible car vous ne pouvez pas ajouter de colonnes à un pseudonyme.

A propos de cette tâche

Pour la version 10.1 ou ultérieure : Vous ne devez pas utiliser cette procédure lors de l'ajout de nouvelles colonnes à la table source si les serveurs participants sont à la version 10.1 ou ultérieure sous z/OS et Linux, UNIX et Windows. Si vous définissez la valeur de la colonne REPL_ADDCOL à Y dans la table IBMQREP_SUBS, lorsque vous ajoutez de nouvelles colonnes à une table, les colonnes sont ajoutées automatiquement à l'abonnement Q, puis à la table cible si elles n'existent pas déjà dans celle-ci. Pour effectuer ce réglage, indiquez REPLICATE ADD COLUMN YES dans la commande ASNCLP CREATE QSUB ou activez la case à cocher **Automatically replicate new columns added to the source table** (Répliquer automatiquement de nouvelles colonnes ajoutées à la table source) lorsque vous créez ou modifiez les propriétés d'un abonnement Q dans le Centre de réplication. Pour plus d'informations, consultez ces rubriques :

-  Enabling replication of ADD COLUMN and SET DATA TYPE operations (Activation de la réplication des opérations ADD COLUMN et SET DATA TYPE)
- «Activation d'une réplication automatique des colonnes nouvellement ajoutées à partir de la table source», à la page 140

Lorsque vous insérez le signal dans le serveur Q Capture, la colonne est ajoutée automatiquement à la table cible si vous ne l'avez pas déjà fait. Si vous souhaitez ajouter plusieurs colonnes à un abonnement Q, insérez un signal pour chaque nouvelle colonne. Vous pouvez ajouter plusieurs colonnes dans une seule transaction. Il est possible d'arrêter Q Capture pendant que vous insérez les signaux qu'il lira depuis le journal lorsqu'il redémarrera.

Laissez les programmes de réplication ajouter automatiquement les nouvelles colonnes à la table source pour garantir qu'elles correspondent aux colonnes de la table source. Les colonnes sont ajoutées à la table cible avec le même type de données, la même caractéristique null et la même valeur par défaut que les colonnes correspondantes de la table source. Vous pouvez spécifier un autre nom de colonne cible si vous utilisez la commande ALTER ADD COLUMN du programme de ligne de commande ASNCLP ou un signal ADDCOL.

Procédure

Pour ajouter des colonnes à un abonnement unidirectionnel, appliquez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande ALTER ADD COLUMN. Par exemple, la commande suivante ajoute la colonne BONUS à l'abonnement DEPARTMENT0001 : <pre>ALTER ADD COLUMN USING SIGNAL (BONUS) QSUB DEPARTMENT0001 USING REPMAP SAMPLE_ASN_TO_TARGET_ASN;</pre>
Tableau de bord de la réplication Q	Dans l'onglet Abonnements, sélectionnez un abonnement Q dans la table et cliquez sur Actions > Ajouter des colonnes .
SQL	Utilisez une invite de commande ou l'un des outils de ligne de commande DB2 pour insérer un signal ADDCOL dans la table IBMQREP_SIGNAL au niveau du serveur Q Capture. Par exemple : <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT TIMESTAMP, 'CMD', 'ADDCOL', 'sous-nom;nom_colonne;nom_colonne_précédente; nom_colonne_suivante', 'P');</pre> <p><i>schéma</i> Identifie le programme Q Capture qui traite l'abonnement Q auquel vous êtes en train d'ajouter une colonne.</p> <p><i>sous-nom;nom_colonne;nom_colonne_précédente;nom_colonne_suivante</i> Nom de l'abonnement Q auquel vous voulez ajouter une colonne et nom de la colonne que vous ajoutez, séparés par un point-virgule. Ces noms sont sensibles à la casse. Il n'est donc pas nécessaire d'utiliser les guillemets pour préserver la casse. Suivez ces exemples :</p> <p>Ajout de colonne dans la table source pour l'abonnement Q et la table cible QSUB1;COL10</p> <p>Ajout de colonne et d'image-avant de la colonne (pour des tables cibles CCD) QSUB1;COL10;XCOL10</p> <p>Ajout de colonne sans image-avant mais avec un nom de colonne cible différent QSUB1;COL10;;TRGCOL10 (placez deux points-virgules (;)) pour indiquer que vous omettez la colonne d'image-avant.)</p>

Après avoir traité le signal, Q Capture commence à enregistrer les changements de la nouvelle colonne lorsqu'il lit les données de journal qui englobent la colonne. Les changements de colonnes qui sont validés après la validation de l'insertion du signal ADDCOL seront répliqués vers la nouvelle colonne dans la table cible. Les lignes qui existaient avant l'ajout de la nouvelle colonne auront une valeur null ou une valeur par défaut pour la nouvelle colonne.

Ajout de colonnes existantes à un abonnement Q (réplication bidirectionnelle ou entre homologues)

Vous pouvez ajouter des colonnes existantes à un abonnement Q bidirectionnel ou entre homologues à partir de la table source lors de l'exécution des programmes de réplication. Si les colonnes n'existent pas et que vos serveurs sont à la version 10.1 ou ultérieure, utilisez une procédure différente.

Avant de commencer

- Les abonnements Q indiquant la table doivent être à l'état A (actif) sur tous les serveurs.
- **Linux UNIX Windows** Si le type de données de la colonne est LONG VARCHAR ou GRAPHIC, la base de données source ou le sous-système doit être configuré avec DATA CAPTURE CHANGES INCLUDE VARCHAR COLUMNS.

Restrictions

- Les colonnes que vous ajoutez doivent avoir pour valeur NULL admis ou être définies comme NOT NULL WITH DEFAULT.
- Si vous ajoutez des colonnes avec des valeurs par défaut, vous devez exécuter l'utilitaire REORG dans la table source avant de démarrer la réplication de la nouvelle colonne. Pour plus d'informations, voir Éviter les erreurs CHECKFAILED lors de l'ajout de colonnes aux tables cible DB2 for z/OS.
- Vous ne pouvez pas modifier la valeur par défaut d'une colonne que vous venez d'ajouter tant que le signal ADDCOL pour cette colonne n'est pas traité.
- Vous ne pouvez pas ajouter plus de 20 colonnes au cours d'un intervalle de validation Q Capture, tel que spécifié par le paramètre **commit_interval**.

A propos de cette tâche

Pour la version 10.1 ou ultérieure : Vous ne devez pas utiliser cette procédure lors de l'ajout de nouvelles colonnes à la table source si les serveurs participants sont à la version 10.1 ou ultérieure sous z/OS et Linux, UNIX et Windows. Si vous définissez la valeur de la colonne REPL_ADDCOL à Y dans la table IBMQREP_SUBS, lorsque vous ajoutez de nouvelles colonnes à une table, les colonnes sont ajoutées automatiquement à l'abonnement Q, puis à la table cible si elles n'existent pas déjà dans celle-ci. Pour effectuer ce réglage, indiquez REPLICATE ADD COLUMN YES dans la commande ASNCLP CREATE QSUB ou activez la case à cocher **Automatically replicate new columns added to the source table** (Répliquer automatiquement de nouvelles colonnes ajoutées à la table source) lorsque vous créez ou modifiez les propriétés d'un abonnement Q dans le Centre de réplication. Pour plus d'informations, consultez ces rubriques :

- **z/OS** Enabling replication of ADD COLUMN and SET DATA TYPE operations (Activation de la réplication des opérations ADD COLUMN et SET DATA TYPE)
- «Activation d'une réplication automatique des colonnes nouvellement ajoutées à partir de la table source», à la page 140

Pour utiliser cette procédure, vous devez tout d'abord modifier une table d'un serveur pour pouvoir ajouter une colonne. Vous insérez ensuite un signal SQL dans le serveur. Lorsque le signal est traité, les versions de la table sur les autres serveurs sont automatiquement modifiées pour ajouter la colonne, sauf si vous l'avez ajoutée manuellement. Ce signal ajoute également la colonne aux définitions des abonnements Q sur tous les serveurs.

Vous pouvez ajouter tout nombre de colonnes dans la table source, à tout moment. Vous pouvez faire cela lorsque les programmes Q Capture et Q Apply sont en cours d'exécution ou sont arrêtés.

Recommandation : Insérez un signal ADDCOL à la fois et émettez une commande COMMIT avant d'insérer un nouveau signal ADDCOL ou d'effectuer toute autre transaction.

Procédure

Ajouter des colonnes à répliquer dans une réplification bidirectionnelle ou dans une réplification entre homologues :

1. Modifiez la table logique sur l'un des serveurs pour ajouter la colonne.
Si l'opération ALTER TABLE qui ajoute la colonne à la table source échoue, tous les abonnements Q dans le groupe de réplification entre homologues seront désactivés.
2. Utilisez l'une des méthodes suivantes pour signaler à Q Capture que vous souhaitez ajouter la colonne à l'abonnement Q pour la table source.

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande ALTER ADD COLUMN. Par exemple, la commande suivante ajoute la colonne BONUS à l'abonnement DEPARTMENT0001 : ALTER ADD COLUMN USING SIGNAL (BONUS) QSUB DEPARTMENT0001 USING REPMAP SAMPLE_ASN_TO_TARGET_ASN;
Tableau de bord de réplification Q	Dans l'onglet Abonnements, sélectionnez un abonnement Q dans la table et cliquez sur Actions > Ajouter des colonnes .

Méthode	Description
SQL	<p>Utilisez une invite de commande ou l'un des outils de ligne de commande DB2 pour insérer un signal ADDCOL dans la table IBMQREP_SIGNAL au niveau du serveur Q Capture. Par exemple :</p> <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'ADDCOL', 'sous-nom;nom_colonne', 'P');</pre> <p><i>schéma</i> Identifie Q Capture sur le serveur où vous avez modifié la table.</p> <p><i>sous-nom;nom_colonne</i> Nom de l'abonnement Q qui part du serveur Q capture où vous avez modifié la table et nom de la colonne que vous ajoutez, séparés par un point-virgule. Ces noms sont sensibles à la casse. Il n'est donc pas nécessaire d'utiliser les guillemets pour préserver la casse.</p>

Soit l'exemple suivant : une configuration entre homologues a trois serveurs, ServerA, ServerB et ServerC, et six abonnements Q, subA2B, subB2A, subA2C, subC2A, subB2C et subC2B pour la table EMPLOYEE.

Vous ajoutez une colonne ADDRESS à la table EMPLOYEE sur le serveur ServerA. Vous insérez ensuite un signal ADDCOL pour l'abonnement Q qui traite les transactions du serveur ServerA vers le serveur ServerB puis spécifiez subA2B;ADDRESS comme nom de l'abonnement Q et nom de colonne. Un signal ADDCOL seulement est requis. Les programmes de réplication ajoutent automatiquement la colonne ADDRESS dans les tables EMPLOYEE des serveurs ServerB et ServerC puis la définition de colonne aux six abonnements Q.

Réplication automatique des opérations ALTER TABLE ALTER COLUMN SET DATA TYPE

A partir de la version 10.1 sous z/OS et Linux, UNIX et Windows, les opérations ALTER TABLE ALTER COLUMN SET DATA TYPE au niveau de la table source sont automatiquement répliquées. Le type de données de la colonne de la table cible correspondante est modifié et la réplication se poursuit normalement.

Conditions requises :

- Les serveurs Q Capture et Q Apply doivent être à la version 10.1 ou plus récente à la fois sous z/OS et Linux, UNIX et Windows. Cette exception s'étend à une colonne VARCHAR/VARGRAPHIC. Cette opération est prise en charge par les versions antérieures.

- **z/OS** Certaines étapes de configurations sont nécessaires sous z/OS. Voir Enabling replication of ADD COLUMN and SET DATA TYPE operations (Activation de la réplication des opérations ADD COLUMN et SET DATA TYPE).
- Vérifiez que l'ID utilisateur qui exécute le programme Q Apply dispose du privilège ALTER sur toutes les tables à modifier.

Dans les situations suivantes, lorsque vous changez le type de données d'une colonne source, vous devez modifier manuellement la colonne cible :

- Le programme Q Apply est à la version 9.7 ou antérieure. Vous devez modifier le type de données de la colonne cible avant de modifier le type de données de la table source.
- La colonne source est mappée vers une expression au niveau de la cible. Vous devez apporter tous les changements requis à l'expression ou à la colonne de table cible sous-jacente.
- La colonne modifiée est une chaîne qui est mappée vers une chaîne plus longue au niveau de la cible pour des besoins de conversion de page de code. Par exemple, supposons une situation dans laquelle le type de données de la colonne source est CHAR(10). La colonne cible est configurée CHAR(40) pour prendre en charge la conversion de page de code. Si la colonne source est changée pour CHAR(20), le changement est répliqué. Toutefois, il n'est pas possible de modifier des types de données chaîne pour une taille plus petite et le changement de CHAR(40) à CHAR(20) doit échouer. Pour éviter cette situation, vous devez modifier manuellement la colonne cible pour CHAR(80) afin de prévenir toute erreur de ligne après une conversion de page de code.

Après la modification de la colonne cible, DB2 change l'état de la table pour REORG PENDING, ce qui requiert une opération REORG avant de pouvoir utiliser la table. Le programme Q Apply appelle la procédure stockée DB2 ADMIN_REVALIDATE_DB_OBJECTS pour retirer la table de l'état REORG PENDING. Les exceptions à cette exigence REORG étendent la longueur des colonnes VARCHAR ou VARGRAPHIC.

Chapitre 8. La réplication Q dans les cibles fédérées

La réplication Q dans les cibles comme Oracle ou Microsoft SQL Server fonctionne comme un scénario où source et cible sont des serveurs DB2.

WebSphere MQ est configuré pour un scénario de réplication de type unidirectionnel entre deux bases de données DB2. Les transactions sont répliquées en utilisant les programmes Q Capture et Q Apply existants.

La principale différence se situe au niveau de Q Apply. Avec une cible autre que DB2, le programme Q Apply s'exécute sur un serveur fédéré, récupérant les changements capturés depuis des files d'attente et les enregistrant dans les tables cible de la base de données relationnelle non DB2 en utilisant les pseudonymes du serveur fédéré.

Lorsque vous configurez la réplication Q pour les cibles fédérées, plusieurs tables de contrôle Q Apply sont créées sur le système cible et sont accessibles au travers de pseudonymes, à l'instar de tables cible. Q Apply enregistre dans ces tables pendant la même unité de travail que les changements apportés aux tables cible. Un ensemble de tables de contrôle est nécessaire pour chaque cible fédérée.

La figure 18 montre les relations entre les programmes de réplication, les serveurs DB2 et le serveur non DB2 dans la réplication Q fédérée.

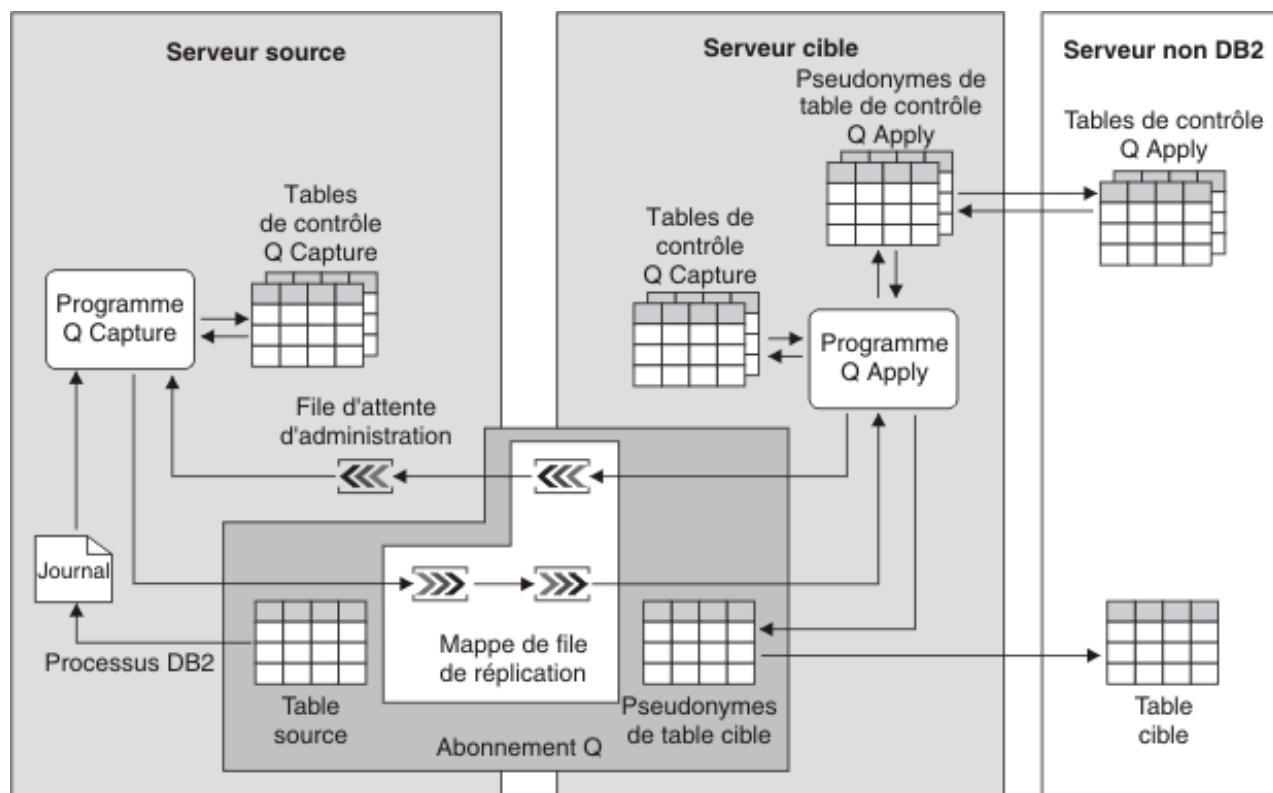


Figure 18. Réplication Q dans un serveur non DB2 par l'intermédiaire d'un serveur fédéré

La prise en charge fédérée suivante de la réplication Q et de la est disponible :

Sources et cibles

La réplication unidirectionnelle est prise en charge à partir de DB2 for z/OS, Linux, UNIX, et Windows dans les tables cible des bases de données relationnelles Oracle, Sybase, Microsoft SQL Server, Informix, ou Teradata qui utilisent des encapsuleurs fédérés définis spécifiquement pour ces cibles. La transformation des données par réplication vers les procédures stockées DB2 qui écrivent dans les pseudonymes est prise en charge. Le programme Q Apply peut charger les données source dans une ou plusieurs tables cible fédérées en parallèle en utilisant les utilitaires EXPORT et IMPORT. Pour les chargements, vous pouvez également indiquer que Q Apply utilise une sélection ODBC afin d'extraire les données de la source ; pour les cibles Oracle, vous pouvez indiquer que Q Apply appelle l'utilitaire SQL*Plus.

La réplication Q prend également en charge l'utilisation d'expressions SQL pour la conversion de données entre les tables source et les tables cible non DB2. Le programme Q Apply effectue la conversion avant d'appliquer les données dans la table cible non DB2 via des pseudonymes sur le serveur fédéré.

Application des données dans les cibles

La réplication Q exige des colonnes d'attributs similaires (par exemple, INTEGER de la table source vers INTEGER du pseudonyme). La réplication Q peut répliquer des colonnes source dans des colonnes de pseudonymes avec différents attributs compatibles comme la réplication à partir de faibles longueurs de données de colonnes à la source vers des longueurs de données plus grandes à la cible (par exemple, la réplication à partir de CHAR(4) à la source vers CHAR(8) à la cible).

Utilitaires

Le moniteur d'alertes de réplication et les utilitaires d'analyse des différences et de synchronisation de table (asntdiff et asntrep) sont pris en charge pour les cibles fédérées. L'utilitaire asntdiff compare la table source avec le pseudonyme et l'utilitaire asntrep met à jour le pseudonyme afin de synchroniser source et cible.

Remarque concernant la terminologie

Dans les rubriques relatives à la réplication Q fédérée, le terme "cible fédérée" fait référence à la base de données Oracle, Sybase, Microsoft SQL Server, Informix ou Teradata qui contient les tables sources. Dans d'autres rubriques fédérées DB2, ces cibles font allusion aux "sources de données". Dans d'autres rubriques de réplication Q, le terme "cible" se rapporte à la base de données DB2 où le programme Q Apply est exécuté et où se situent les tables cible. Cette base de données contient les pseudonymes dans la réplication Q fédérée. La figure 19, à la page 151 illustre ces relations.

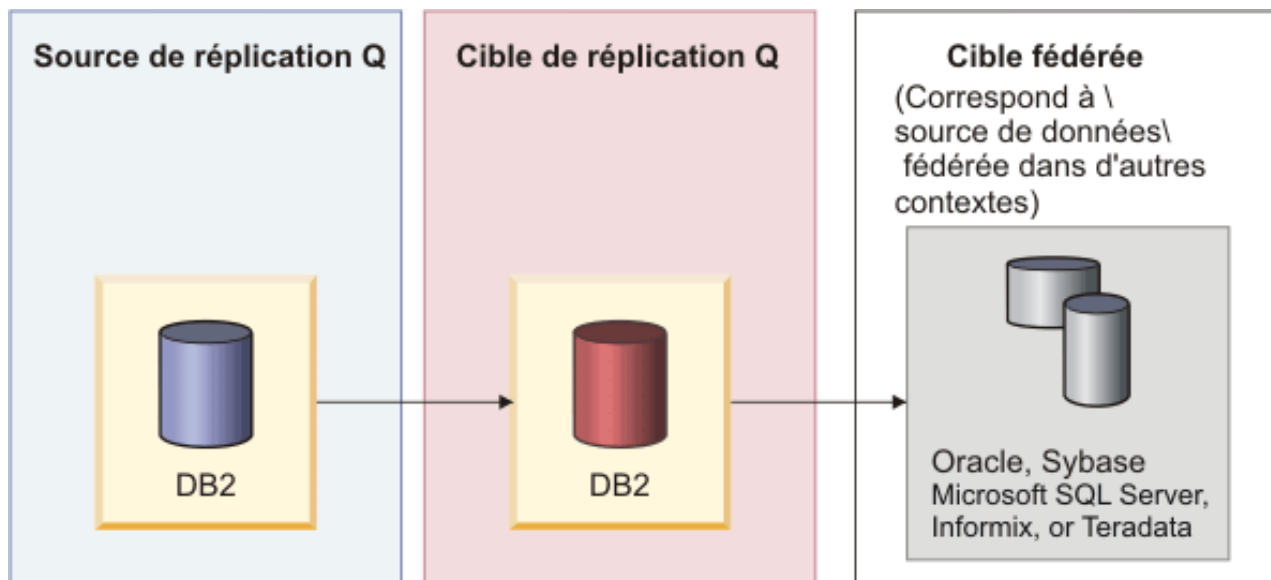


Figure 19. Sources et cibles dans la réplication Q fédérée

Paramétrage de la réplication Q vers des cibles fédérées

La configuration de la réplication Q pour des cibles fédérées implique la configuration du serveur fédéré sur lequel le programme Q Apply s'exécute, la création d'objets WebSphere MQ, d'objets fédérés et d'objets de réplication Q.

Restrictions générales

- L'utilisation des commandes `asntdiff` et `asntrep` implique que les types de données de la table source DB2 et du pseudonyme sur le serveur fédéré où est exécuté le programme Q Apply soient identiques.
- Si vous utilisez le signal `ADDCOL` pour ajouter une colonne à un abonnement Q existant, cette nouvelle colonne doit déjà figurer dans la table cible et le pseudonyme correspondant (vous ne pouvez pas ajouter de colonne à un pseudonyme).
- Assurez-vous que les pages de codes de la base de données source, de la base de données DB2 fédérée et de la base de données cible non-DB2 correspondent.

Restrictions sur les types de données

- La réplication des valeurs LOB (large object) est prise en charge uniquement pour les cibles Oracle, ces dernières devant utiliser l'encapsuleur `NET8`.
- Pour répliquer les types de données `GRAPHIC`, `VARGRAPHIC` ou `DBCLOB`, vous devez utiliser un serveur et un client Oracle version 9 ou supérieure, tout comme votre mappage serveur.
- La réplication des types de données `LONG VARGRAPHIC` n'est pas prise en charge.
- Pour Sybase, Microsoft SQL Server et Informix, si le type de données de colonne de la table source est `LONG VARCHAR`, le pseudonyme est créé avec le type de données `VARCHAR(32672)`. La longueur autorisée pour un type `LONG VARCHAR` étant supérieure à 32672 caractères, certaines données peuvent être tronquées.

Procédure

Pour configurer la réplication Q vers des cibles fédérées :

1. Configurez le serveur fédéré.
2. Créez des objets fédérés.
3. Créez des objets WebSphere MQ.
4. Créez des tables de contrôle.
5. Créez des abonnements Q.

La figure 20, à la page 153 affiche les étapes du paramétrage de la réplication Q fédérée.

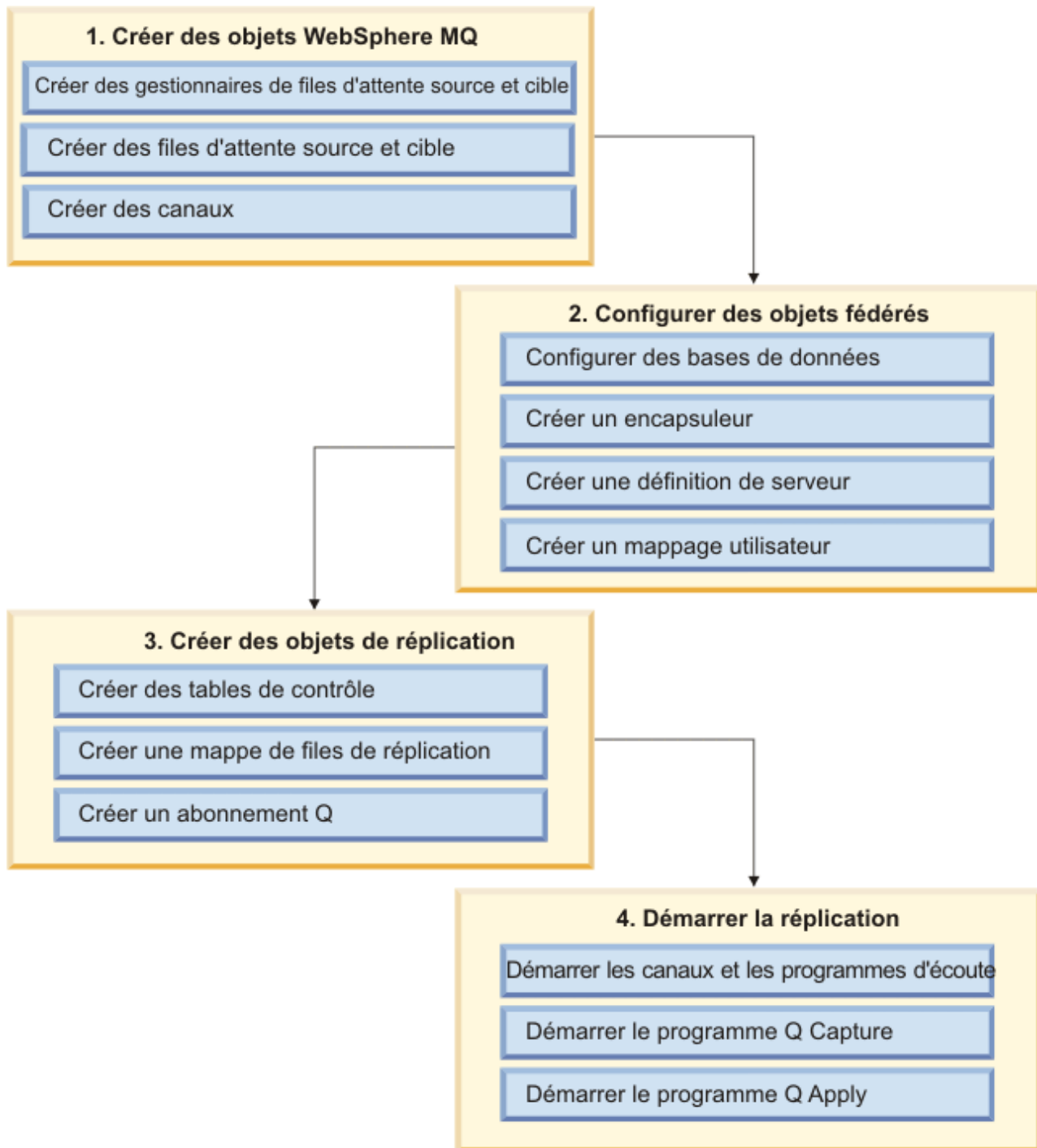


Figure 20. Présentation des étapes de paramétrage de la réplication Q fédérée

Configuration du serveur fédéré pour la réplication Q

Avant de répliquer vers une cible fédérée, vous devez configurer l'instance et la base de données DB2 sur le système d'exécution du programme Q Apply.

Avant de commencer

Assurez-vous que les pages de codes de la base de données source, de la base de données DB2 fédérée et de la base de données non DB2 correspondent.

Procédure

Pour configurer le serveur fédéré en vue d'une réplication Q :

1. Activez la prise en charge de la fédération dans l'instance DB2 sur laquelle le programme Q Apply est exécuté, en appliquant l'un des procédés suivants :

Méthode	Description
Centre de contrôle	Utilisez la fenêtre Configuration DBM. Pour ouvrir cette fenêtre, dans l'arborescence d'objets, cliquez avec le bouton droit de la souris sur l'instance DB2 qui contient la base de données Q Apply, puis cliquez sur Configuration des paramètres . Sous l'en-tête Environnement , cliquez sur Fédéré , puis sur le bouton "..." pour activer la prise en charge de la fédération.
Commande UPDATE DBM CFG	Vérifiez que vous êtes sur l'instance qui contient le serveur Q Apply, puis saisissez la commande suivante : UPDATE DBM CFG USING FEDERATED YES

Vous devez redémarrer l'instance pour que la modification prenne effet.

2. Paramétrez et testez le serveur Oracle, Sybase ou Microsoft SQL, ou le fichier de configuration client Informix sur le système d'exécution du programme Q Apply. Pour plus de détails, reportez-vous à l'une des rubriques suivantes :
 - Paramétrage et test du fichier de configuration client Oracle
 - Paramétrage et test du fichier de configuration client Sybase
 - Préparation du serveur fédéré pour l'accès aux sources de données du serveur Microsoft SQL
 - Configuration et test du fichier de configuration client Informix
3. Définissez les variables d'environnement pour la connexion au serveur Oracle, Sybase, Microsoft SQL ou Informix dans le fichier db2dj.ini sur le serveur fédéré où le programme Q Apply est exécuté. Pour plus de détails, reportez-vous à l'une des rubriques suivantes :
 - Définition des variables d'environnement Oracle
 - Définition des variables d'environnement Sybase
 - Définition des variables d'environnement Microsoft SQL Server
 - Définition des variables d'environnement Informix
 - Définition des variables d'environnement Teradata
4. **Requis pour les cibles Sybase :** Liez les modules Q Apply manuellement avec le isolement CS (lecture non reproductible).

Création d'objets fédérés pour la réplication Q

Pour que le programme Q Apply puisse communiquer et échanger des données avec la cible fédérée, vous devez créer ou enregistrer des encapsuleurs, des définitions de serveur et des mappages utilisateur sur le serveur fédéré.

Avant de commencer

- L'instance DB2 qui contient le serveur Q Apply doit être configuré pour prendre la fédération en charge.
- Les encapsuleurs utilisés pour la réplication sur des cibles fédérées doivent autoriser l'accès en écriture sur la cible fédérée.

Restrictions

La réplication Q ne prend pas en charge la réplication vers les cibles fédérées au moyen d'un encapsuleur générique tel qu'ODBC.

Procédure

Pour créer des objets fédérés pour Q Replication :

1. Enregistrez l'encapsuleur approprié à votre base de données Oracle, Sybase, Microsoft SQL Server ou Informix. Reportez-vous à l'une des rubriques suivantes :
 - Enregistrement de l'encapsuleur Oracle
 - Enregistrement de l'encapsuleur Sybase
 - Enregistrement de l'encapsuleur Microsoft SQL Server
 - Enregistrement de l'encapsuleur Informix
 - Enregistrement de l'encapsuleur Teradata
2. Enregistrez la définition de serveur correspondant à votre cible Oracle, Sybase, Microsoft SQL Server ou Informix. Reportez-vous à l'une des rubriques suivantes :
 - Enregistrement des définitions de serveur pour une source de données Oracle
 - Enregistrement des définitions de serveur pour une source de données Sybase
 - Enregistrement des définitions de serveur pour une source de données Microsoft SQL Server
 - Enregistrement des définitions de serveur pour une source de données Informix
 - Enregistrement des définitions de serveur pour une source de données Teradata
3. Créez un mappage utilisateur entre l'ID utilisateur figurant sur le serveur fédéré où est exécuté le programme Q Apply et l'ID utilisateur figurant sur le serveur Oracle, Sybase, Microsoft SQL Server ou Informix. Reportez-vous à l'une des rubriques suivantes :
 - Création des mappages utilisateur pour une source de données Oracle
 - Création d'un mappage utilisateur pour une source de données Sybase
 - Création d'un mappage utilisateur pour une source de données Microsoft SQL Server
 - Création d'un mappage utilisateur pour une source de données Informix
 - Création d'un mappage utilisateur pour une source de données Teradata

L'ID utilisateur du mappage doit disposer des droits d'accès suivants :

- Création de tables dans la base de données cible (tables de contrôle et table cible si vous le souhaitez).
- Droits d'accès SELECT, INSERT, UPDATE et DELETE pour les tables de contrôle et la table cible.

Restriction : Le Centre de réplication et le programme de ligne de commande ASNCLP ne prennent pas en charge la création des tables de contrôle ni des tables cible dans les bases de données Oracle si la validation en deux phases du mappage de serveur est activée.

Création de tables de contrôle Q Apply pour la réplication Q fédérée

Pour être en mesure de répliquer des données dans cible fédérée, vous devez créer des tables de contrôle pour stocker les informations relatives aux abonnements Q, aux files d'attente de messages, aux paramètres opérationnels et aux préférences utilisateur.

Avant de commencer

- Vous devez configurer un accès fédéré en créant des encapsuleurs, des définitions de serveur et des mappages utilisateur. Les outils d'administration de la réplication fonctionneront avec la cible Oracle, Sybase, Microsoft SQL Server, Informix ou Teradata si vous utilisez des encapsuleurs.
- Par défaut, l'ID utilisateur distant est utilisé comme schéma pour les tables de contrôle Q Apply créées dans la base de données cible Oracle, Sybase, Microsoft SQL Server, Informix ou Teradata. L'ID utilisateur doit disposer des droits requis pour créer des objets dans ce schéma.

A propos de cette tâche

Pour les cibles fédérées, certaines tables de contrôle Q Apply sont créées dans la base de données cible et sont accessibles au travers de pseudonymes, à l'instar de tables cible. Le reste des tables de contrôle est créé dans le serveur fédéré où s'exécute le programme Q Apply. Le tableau 12 montre l'emplacement des tables de contrôle.

Tableau 12. Emplacement des tables de contrôle pour la réplication Q fédérée

Tables dans le serveur fédéré	Tables du serveur cible non DB2
<ul style="list-style-type: none">• IBMQREP_APPLYENQ• IBMQREP_APPLYTRACE• IBMQREP_APPLYMON• IBMQREP_APPLYPARMS	<ul style="list-style-type: none">• IBMQREP_DONEMSG• IBMQREP_EXCEPTIONS• IBMQREP_RECVQUEUES• IBMQREP_SAVERI• IBMQREP_SPILLEDROW• IBMQREP_SPILLQS• IBMQREP_TRG_COLS• IBMQREP_TARGETS

Informix : le type de données de la colonne MQMSGID des tables de contrôle IBMQREP_DONEMSG et IBMQREP_SPILLEDROW est modifié au profit de BYTE sur Informix afin d'empêcher le serveur fédéré de tronquer les données binaires. En outre, aucune clé primaire n'est définie pour cette table car Informix ne permet pas la création d'index sur les données binaires.

Procédure

Pour créer des tables de contrôle Q Apply pour la réplication Q fédérée :

Utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande CREATE CONTROL TABLES FOR. Spécifiez le mot clé FEDERATED.</p> <p>Vous avez également la possibilité d'utiliser le mot clé RMT SCHEMA pour spécifier un schéma pour les tables de contrôle sur la cible fédérée. La valeur par défaut correspond à l'ID utilisateur distant. Vous pouvez aussi définir l'espace table (Oracle), le segment (Sybase), le groupe de fichiers (SQL Server) ou le dbspace (Informix) où ces tables de contrôle distantes seront créées.</p> <p>A titre d'exemple, les commandes suivantes définissent l'environnement et créent des tables de contrôle Q Apply pour la réplication dans une cible Oracle ORACLE_TARGET utilisant un serveur fédéré FED_DB et un ID utilisateur distant FED_ASN :</p> <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER TARGET TO DB FED_DB NONIBM SERVER ORACLE_TARGET; SET QMANAGER QM2 FOR APPLY SCHEMA; SET APPLY SCHEMA ASN; CREATE CONTROL TABLES FOR APPLY SERVER IN FEDERATED RMT SCHEMA FED_ASN;</pre>
Centre de réplication	<p>Utilisez l'assistant Créer des tables de contrôle Q Apply. Pour ouvrir cet assistant, cliquez avec le bouton droit sur le dossier Serveurs Q Apply et cliquez sur Créer des tables de contrôle Q Apply .</p> <p>Page d'accueil</p> <p>Pour indiquer l'emplacement des tables de contrôle sur le serveur Q Apply ou un serveur non DB2, cliquez sur Personnalisé.</p> <ul style="list-style-type: none"> • Pour le serveur Q Apply, les tables de contrôle sont créées dans un espace table. Vous pouvez spécifier un espace table existant ou créer un nouvel espace table. • Au niveau de la cible fédérée, les tables de contrôle sont créées dans l'espace table (Oracle), le segment (Sybase), le groupe de fichiers (Microsoft SQL Server) ou le dbspace (Informix) par défaut, mais vous pouvez aussi définir un espace table, un segment, un groupe de fichiers ou un dbspace existant. <p>Page Serveur</p> <p>Sélectionnez un serveur fédéré.</p> <p>Page Tables cible</p> <p>Indiquez que les tables cible résident dans une base de données relationnelle non DB2 mappée vers le serveur Q Apply et vérifiez :</p> <ul style="list-style-type: none"> • le nom du serveur (la définition de serveur du serveur relationnel non-DB2 mappé vers le serveur fédéré DB2). • Le schéma distant dans lequel les tables de contrôle Q Apply seront créées dans la base de données Oracle, Sybase, Microsoft SQL Server, Informix ou Teradata (la zone Schéma affiche l'ID utilisateur distant pour ces cibles, mais vous pouvez modifier cette valeur).

Création d'abonnements Q pour la réplication Q fédérée

Un abonnement Q pour une réplication Q fédérée mappe la table DB2 qui contient vos données source dans une copie de cette table dans la cible fédérée. Lorsque vous créez un abonnement Q, vous définissez une mappe de files d'attente, des options de table cible et d'autres préférences. Vous créez un abonnement Q pour chaque table que vous souhaitez répliquer.

Avant de commencer

Pour répliquer dans des cibles Informix, vous devez activer la consignation de transaction avec l'option **buffered_log**.

Restrictions

- La réplication multidirectionnelle n'est pas prise en charge pour les cibles fédérées.
- Seules les procédures stockées DB2 écrivant dans des pseudonymes sont prises en charge. La procédure stockée ne peut pas écrire à la fois dans des pseudonymes et dans des tables DB2 locales car une validation en deux phases ne peut pas être prise en charge.
- Les vues ou les procédures stockées dans une base de données Oracle, Sybase, Microsoft SQL Server, Informix ou Teradata ne sont pas prises en charge en tant que cibles.
- Si vous souhaitez que le programme Q Apply charge la table cible, vous devez spécifier les utilitaires EXPORT et IMPORT pour toutes les cibles sauf Oracle (pour lequel vous pouvez utiliser SQL*Plus). Avant d'utiliser cette fonction de chargement, les pseudonymes qui font référence aux tables cible Teradata doivent être vides. Les pseudonymes qui font référence aux tables dans les versions de base de données Oracle, Microsoft SQL Server, Informix, et Sybase qui prennent en charge les opérations TRUNCATE n'ont pas besoin d'être vides car Q Apply lance une opération TRUNCATE sur la table cible. Les opérations Truncate ne sont pas prises en charge si la table cible est assortie de contraintes d'intégrité référentielle.

Cibles Oracle : Pour utiliser SQL*Plus, vous devez créer un fichier de mots de passe à l'aide de la commande **asnpwd** dans le répertoire spécifié par le paramètre **apply_path** ou le répertoire depuis lequel Q Apply est appelé avec les valeurs suivantes pour ces mots-clés :

- **alias** : L'entrée Oracle `tnsnames.ora` fait référence au serveur Oracle (nom identique utilisé avec l'option `NODE` de la commande `CREATE SERVER` pour configurer la fédération).
- **id** : ID utilisateur distant permettant de se connecter à la base de données Oracle.
- **password** : mot de passe permettant de se connecter à la base de données Oracle.

Le fichier doit s'appeler `asnpwd.aut` par défaut. Avant de commencer l'abonnement Q, testez la connectivité avec cette commande : `$> sqlplus id/mot de passe@alias`.

- La réplication dans les tables cible fédérées présentant des contraintes d'intégrité référentielle est prise en charge seulement si vous définissez manuellement les contraintes sur les pseudonymes correspondants. Le programme `ASNCLP` ou le Centre de réplication ne crée pas automatiquement ces contraintes lorsque les outils créent des pseudonymes. De plus, le programme Q Apply ne supprime

pas les contraintes d'intégrité référentielle sur les pseudonymes pendant le processus de chargement pour les récupérer après.

Recommandation : Utilisez l'option "pas de chargement" pour les pseudonymes présentant des contraintes d'intégrité référentielle et chargez la table cible en dehors des outils d'administration de la réplication.

- Pour les pseudonymes cible comportant plusieurs index, l'attribut BEFORE_VALUES pour l'abonnement Q doit être Y et la valeur CHANGED_COLS_ONLY doit être N dans la table IBMQREP_SUBS.

Procédure

Pour créer des abonnements Q en vue d'une réplication Q fédérée :

Utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande CREATE QSUB pour la réplication unidirectionnelle. Pour TARGET NAME indiquez la table cible non DB2. Spécifiez le mot clé FEDERATED. Vous pouvez spécifier un nom de pseudonyme et un propriétaire si vous souhaitez modifier la valeur par défaut (facultatif). Par exemple, les commandes suivantes créent un abonnement Q présentant ces caractéristiques :</p> <ul style="list-style-type: none"> • Le serveur source est SAMPLE. • Le serveur fédéré où s'exécute le programme Q Apply (mot de passe TARGET) est FED_DB. • Le serveur cible non DB2 (mot de passe NONIBM SERVER) est ORACLE_TARGET. • La mappe de files de réplication (mot de passe REPLQMAP) est SAMPLE_ASN_TO_FED_DB_ASN. • Le nom de l'abonnement Q (mot de passe SUBNAME) est FEDSUB. • La table cible sur la base de données Oracle est EMPLOYEE. • Le pseudonyme sur le serveur Q Apply fédéré qui pointe vers la table EMPLOYEE (mot de passe NICKNAME) est EMPNICKNAME • L'abonnement Q définit une phase de chargement manuel (E) (mot de passe HAS LOAD PHASE) <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET SERVER TARGET TO DB FED_DB NONIBM SERVER ORACLE_TARGET; SET CAPTURE SCHEMA SOURCE ASN; SET APPLY SCHEMA ASN; SET QMANAGER QM1 FOR CAPTURE SCHEMA; SET QMANAGER QM2 FOR APPLY SCHEMA; CREATE QSUB USING REPLQMAP SAMPLE_ASN_TO_FED_DB_ASN (SUBNAME FEDSUB TARGET NAME EMPLOYEE FEDERATED NICKNAME EMPNICKNAME OPTIONS HAS LOAD PHASE E);</pre>

Méthode	Description
Centre de réplication	<p>Utilisez l'assistant Créer des abonnements Q. Pour ouvrir l'assistant, développez le schéma Q Capture ou Q Apply approprié dans l'arborescence des objets, cliquez avec le bouton droit de la souris sur le dossier Abonnements Q , et sélectionnez Créer.</p> <p>Page Serveurs Spécifiez le serveur fédéré sur lequel le programme Q Apply est exécuté en tant que serveur cible. Le programme Q Apply met à jour un pseudonyme sur ce serveur qui est mappé vers une table sur la base de données relationnelle non DB2.</p> <p>Page cible Définissez le type de cible dans laquelle vous souhaitez répliquer :</p> <p>Table de la base de données non DB2 La table sera mise à jour en utilisant un pseudonyme. Vous pouvez laisser le Centre de réplication créer une nouvelle table ou spécifier une table existante.</p> <p>Procédure stockée DB2 Une procédure stockée vous permet de manipuler des données source avant qu'elles ne soient appliquées au pseudonyme. La procédure stockée doit déjà exister sur le serveur Q Apply et doit seulement écrire dans les pseudonymes.</p> <p>Pour plusieurs abonnements Q : si vous spécifiez plus d'une table source, la page des tables cible affiche le profil qui doit être utilisé pour les tables cible non DB2, les index et espaces table ou segments ainsi que pour les pseudonymes cible sur le serveur fédéré. Cliquez sur Modifier pour ouvrir la fenêtre Gérer le profil d'objet cible et changez les noms.</p> <p>Page des lignes et des colonnes Utilisez le champ Mappage de colonne si vous avez besoin de modifier le mappage par défaut des colonnes source vers les colonnes dans la table cible non DB2. La fenêtre Mappage de colonne affiche les mappages de type de données par défaut entre le pseudonyme sur le serveur fédéré et la table cible non DB2 et valide dans le cas où une colonne source peut être mappée vers une colonne cible.</p>

Chapitre 9. Réplication Q à partir de sources Oracle

La réplication Q fournit une capture de journal de tables source Oracle combinée au haut débit et à la faible latence de la messagerie WebSphere MQ et à un programme Q Apply hautement parallèle.

Une utilisation fréquente de cette solution est la réplication de données à partir d'un traitement des transactions en ligne (IOLTP) sur Oracle dans un serveur en lecture seule. Cette approche permet les requêtes en temps réel des données de gestion live pour la génération de rapports, tout en réduisant le temps système et en évitant la contention au niveau du serveur OLTP.

La figure suivante montre la fonctionnalité Oracle LogMiner convertissant des enregistrements de journaux des tables source en instructions SQL, que le programme Q Capture transforme ensuite en messages WebSphere MQ pour les transmettre au programme Q Apply. Une mappe de files d'attente de réplication indique les files d'attente de messages qui sont utilisées pour transmettre les transactions, et les abonnements Q mappent les tables source Oracle aux tables cible DB2.

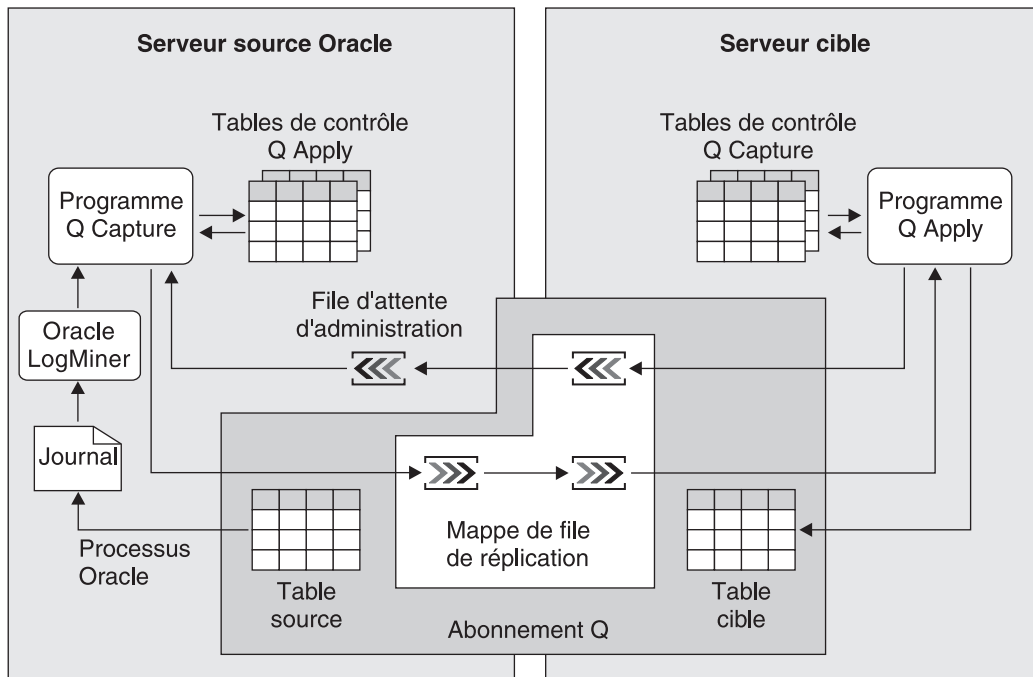


Figure 21. Présentation de haut niveau de la réplication Q à partir d'une source Oracle

Configuration de la réplication Q à partir de sources Oracle

Le paramétrage de la réplication Q à partir de sources Oracle implique la configuration de la base de données source Oracle, la création d'objets WebSphere MQ, la configuration des serveurs Q Capture et Q Apply, et la création d'objets de réplication Q.

Procédure

Pour paramétrer la réplication Q à partir de sources Oracle :

1. Configurez la source Oracle.
2. Créez des objets WebSphere MQ.
3. Créez des tables de contrôle.
4. Paramétrez le chargement de la table cible.
5. Créez des abonnements Q.
6. Démarrez le programme Q Capture
7. Démarrez le programme Q Apply

La figure 22, à la page 163 affiche les étapes du paramétrage de la réplication Q à partir des sources Oracle.

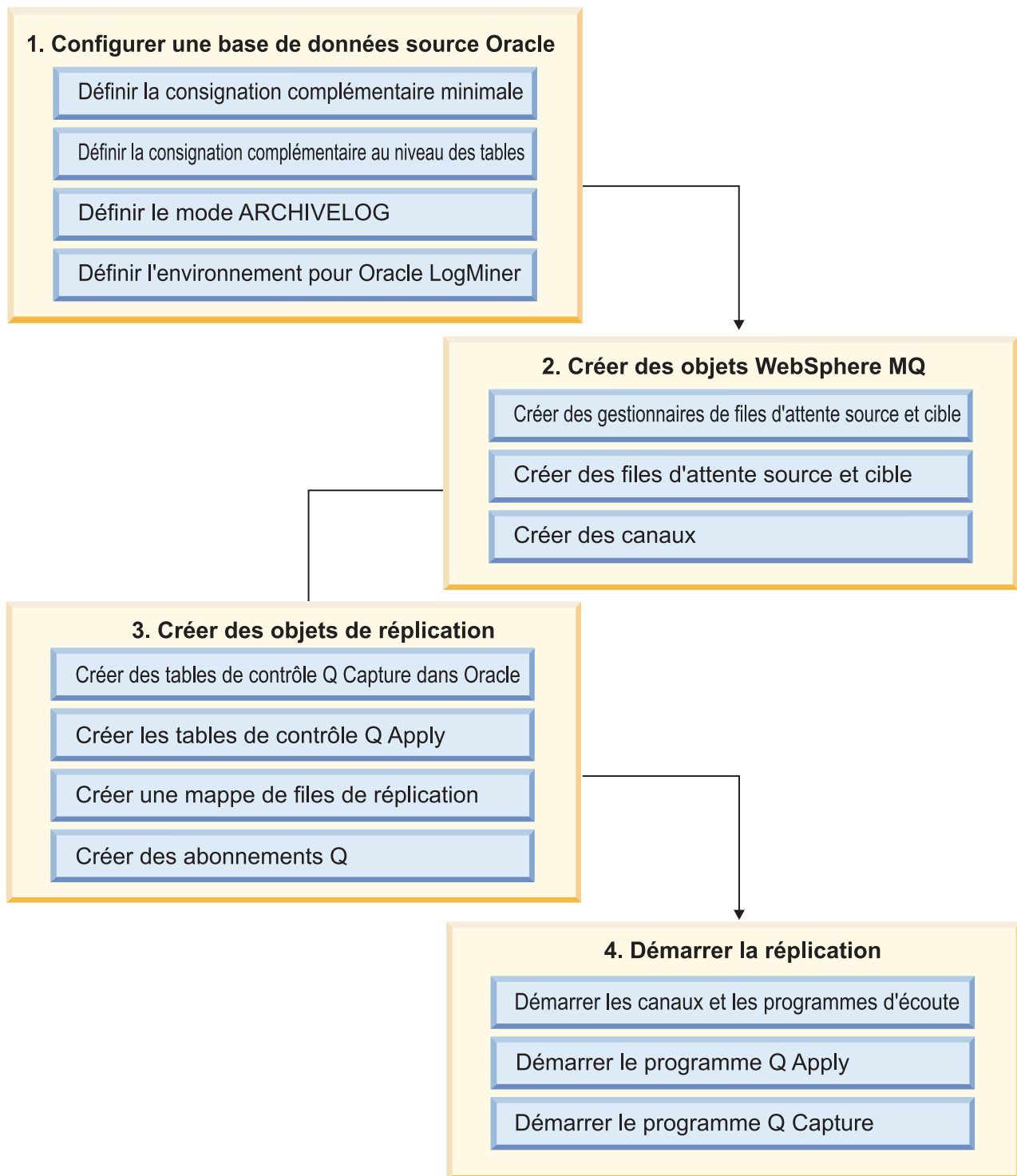


Figure 22. Présentation des étapes du paramétrage de la réplication Q à partir d'Oracle

Support client Oracle pour la réplication Q

Si vous exécutez Q Capture pour les sources Oracle, vous devez utiliser les bibliothèques client Oracle adéquates pour votre système d'exploitation.

Un client Oracle complet n'est pas tenu d'exécuter le programme Q Capture avec une base de données source Oracle. Vous pouvez utiliser ce qu'Oracle définit comme un Client instantané, ce qui est un petit fichier compressé qui comprend les fichiers DLL essentiels ou les bibliothèques partagées requises par une application client Oracle.

Pour utiliser le Client instantané avec Q Capture pour Oracle :

1. Extrayez le fichier compressé du client instantané dans un répertoire dédié (ne pas mélanger avec le répertoire client Oracle complet).
2. Veillez à ce que le répertoire du client instantané soit inclus dans l'une des variables d'environnement suivantes : PATH (Windows), LIBPATH (UNIX), LD_LIBRARY_PATH (Linux ou Solaris), ou SHLIB_PATH (HP-UX).
3. Si vous avez un autre client installé (éventuellement depuis une installation Oracle) qui ne correspond pas à la version requise ci-dessous, ce client ne doit pas être mélangé avec le Client instantané dans les variables d'environnement, ou vous devez vous assurer que le chemin d'accès au répertoire du Client instantané est créé avant le chemin d'accès vers le répertoire du Client Oracle installé.

Les tables ci-dessous dressent la liste des systèmes d'exploitation pris en charge, ainsi que le niveau de client Oracle requis pour le programme Q Capture. Le client Oracle n'est pas obligé de correspondre au serveur Oracle, mais la version du logiciel du serveur de base de données doit être égale ou supérieure à la version du logiciel client.

InfoSphere Replication Server version 9.7 groupe de correctifs 3a et versions antérieures

Le tableau suivant dresse la liste des systèmes d'exploitation pris en charge et des clients obligatoires lorsque vous utilisez InfoSphere Replication Server version 9.7 groupe de correctifs 3a et les versions antérieures pour effectuer une réplication à partir des bases de données Oracle.

Tableau 13. Systèmes d'exploitation pris en charge et version de la bibliothèque client Oracle obligatoire pour InfoSphere Replication Server version 9.7 groupe de correctifs 3a et les versions antérieures

Système d'exploitation	Version de bibliothèque client Oracle
AIX 5.3	10.2.0.1 client instantané
Linux 390 64 bits	10.2.0.1 client instantané
Solaris 64 bits	10.2.0.1 client instantané
Linux AMD 64 bits	10.2.0.3 client instantané corrigé avec version 10.2.0.3.0 g++ 3.4.3 RHEL AS 4.0 64 bits (x86_64)
Windows 32 bits	10.2.0.3 client instantané corrigé avec la version 10.2.0.3.0 VS2005 OCCI
HP IPF 64 bits	10.2.0.1 client instantané

WebSphere Replication Server version 9.5

Le tableau suivant dresse la liste des systèmes d'exploitation pris en charge et des clients obligatoires lorsque vous utilisez WebSphere Replication Server version 9.5 pour effectuer une réplication à partir des bases de données Oracle.

Tableau 14. Systèmes d'exploitation pris en charge et version de la bibliothèque client Oracle obligatoire pour WebSphere Replication Server version 9.5

Système d'exploitation	Version de bibliothèque client Oracle
AIX 5.3	10.2.0.1 client instantané
Linux 390 64 bits	10.2.0.1 client instantané
Solaris 64 bits	10.2.0.1 client instantané
Linux AMD 64 bits	10.2.0.1 client instantané
Windows 32 bits	10.2.0.3 client instantané corrigé avec la version 10.2.0.3.0 VS2005 OCCI
HP IPF 64 bits	10.2.0.1 client instantané

Configuration de la source Oracle

Pour permettre la capture des transactions à partir de la base de données source Oracle, vous devez définir les options de consignation requises, la variable d'environnement IBMQREPLPATH et permettre au programme Q Capture d'utiliser l'utilitaire Oracle LogMiner.

Configuration d'une base de données source Oracle pour travailler avec un programme Q Capture

Vous devez activer la consignation des archives, la consignation complémentaire minimale, la consignation complémentaire au niveau des tables sur une base de données source Oracle avant de pouvoir démarrer le programme Q Capture.

A propos de cette tâche

La liste suivante décrit chaque exigence de configuration :

Consignation des archives

La base de données source Oracle doit s'exécuter en mode ARCHIVELOG. Ce mode donne à la fonctionnalité LogMiner l'accès à tous les journaux redo pouvant être nécessaires pour mettre les serveurs source et cible en convergence. Le programme Q Capture utilise la fonctionnalité LogMiner avec l'option CONTINUOUS_MINE. Pour exécuter une extraction continue à partir des journaux Oracle en utilisant LogMiner, la consignation d'archives doit être activée. Lors de l'initialisation, si le programme Q Capture détecte que la base de données s'exécute en mode NOARCHIVELOG, le programme ne démarrera pas.

Consignation complémentaire minimale

La fonctionnalité LogMiner exige que la consignation complémentaire soit activée. Par défaut, Oracle ne fournit pas de consignation complémentaire. Vous devez activer au moins la consignation complémentaire minimale avant que tout journal redo soit généré pour être utilisé par la réplication.

Consignation complémentaire au niveau des tables

Le programme Q Capture exige que la consignation complémentaire au niveau des tables soit activée pour toute table source Oracle faisant partie

d'un abonnement Q. Les données ne sont pas capturées pour une table tant que la consignation complémentaire au niveau des tables n'est pas activée.

Procédure

Pour configurer une base de données source Oracle pour travailler avec le programme Q Capture, procédez comme suit :

- Veillez à ce que la consignation d'archives soit activée. Pour déterminer le mode de consignation de la base de données, (ARCHIVELOG or NOARCHIVELOG), utilisez l'instruction suivante :

```
SELECT LOG_MODE FROM SYS.V$DATABASE;
```

- Définissez la consignation complémentaire minimale en utilisant l'instruction suivante :

```
ALTER DATABASE ADD SUPPLEMENTAL LOG DATA;
```

Pour vérifier quelle consignation complémentaire est activée, interrogez la vue V\$DATABASE :

```
SELECT SUPPLEMENTAL_LOG_DATA_MIN FROM V$DATABASE;
```

Si la requête renvoie une valeur YES ou IMPLICIT, la consignation complémentaire minimale est activée.

- Pour chaque table source table, activez la consignation complémentaire au niveau des tables en utilisant l'instruction suivante :

```
ALTER TABLE propriétaire_table.nom_table ADD SUPPLEMENTAL LOG DATA (ALL) COLUMNS;
```

Pour plus d'informations sur l'activation de la consignation des archives, reportez-vous au chapitre "Gestion des journaux redo archivés" dans le manuel *Oracle Database Administrator's Guide*. Pour plus d'informations sur l'activation de la consignation complémentaire, voir "Consignation complémentaire" dans *Oracle Database Utilities*.

Comment le programme Q Capture travaille avec la fonctionnalité Oracle LogMiner

Le programme Q Capture utilise la fonctionnalité Oracle LogMiner pour extraire les données modifiées des tables source Oracle avant de transformer les données en messages WebSphere MQ pour la réplication ou la publication.

La figure suivante montre les principaux objets et application impliqués dans la capture des données à partir des tables source Oracle.

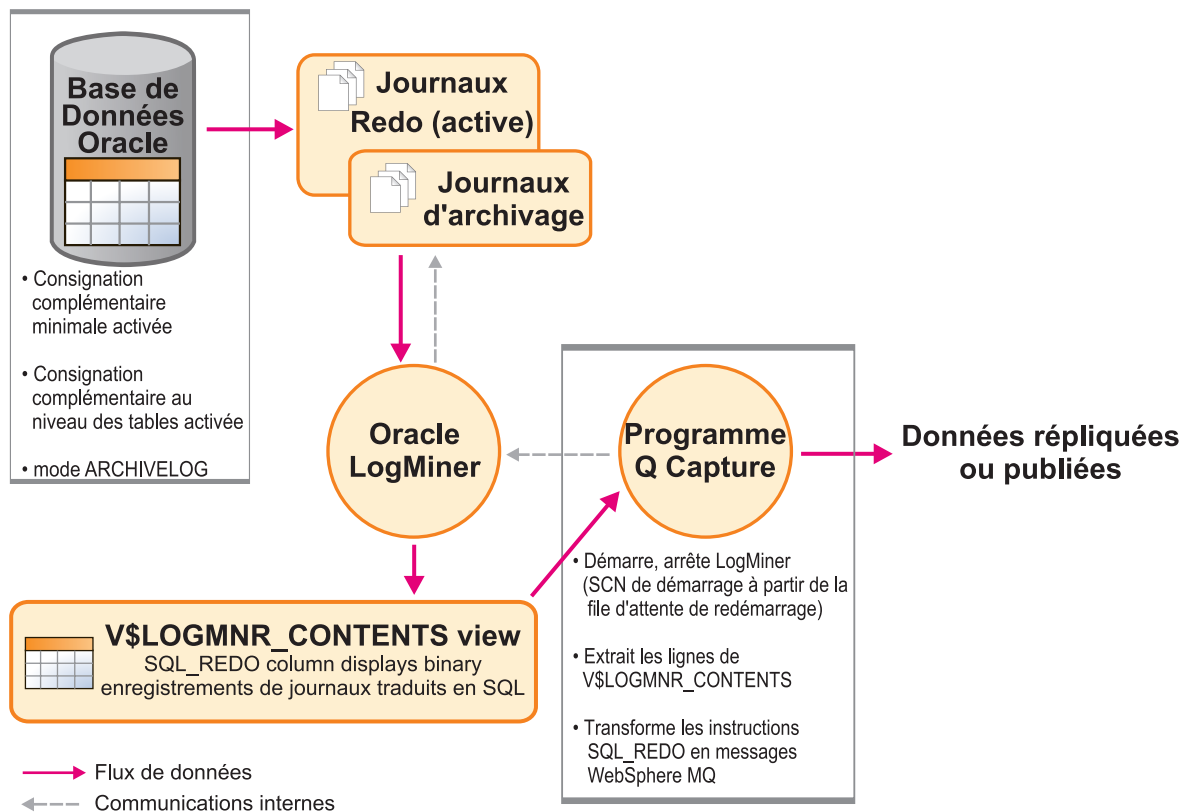


Figure 23. Comment un Q Capture travaille avec LogMiner pour extraire les données modifiées

La liste suivante décrit comment ces objets fonctionnent ensemble :

Journal redo

Le journal redo Oracle se compose de fichiers préalloués qui contiennent toutes les modifications apportées à la base de données. Pour la réplication Q ou la publication d'événements, la base de données doit fonctionner en mode ARCHIVELOG afin que le gestionnaire de base de données n'écrase jamais les journaux redo nécessaires à la réplication ou à la publication.

Fonctionnalité LogMiner

La fonctionnalité se connecte à la base de données source Oracle, lit les journaux d'archive et les journaux redo actifs, et présente les données d'enregistrement de journal dans une pseudo-vue appelée V\$LOGMNR_CONTENTS. Les données sont reproduites dans un format SQL utilisé par le programme Q Capture pour recréer les modifications.

Dictionnaire

LogMiner utilise un dictionnaire pour traduire les enregistrements de journaux binaires en instructions SQL en texte normal. La réplication Q et la publication d'événements spécifient un dictionnaire en ligne. Un dictionnaire en ligne reflète toujours le schéma de base de données en cours. Pour plus d'informations, voir «Comment Q Capture gère les modifications des tables source Oracle», à la page 177.

V\$LOGMNR_CONTENTS

LogMiner crée la vue V\$LOGMNR_CONTENTS en réponse aux requêtes SQL de Q Capture. Q Capture utilise les instructions SQL régénérer de V\$LOGMNR_CONTENTS pour créer des messages WebSphere MQ pour la réplication ou la publication.

Q Capture, programme

Le programme Q Capture démarre, arrête et envoie des requêtes à la fonctionnalité LogMiner. Q Capture administre la fonctionnalité en utilisant deux modules Oracle PL/SQL : DBMS_LOGMNR_D pour créer le dictionnaire en ligne, et DBMS_LOGMNR pour démarrer et arrêter la fonctionnalité, ajouter et supprimer des fichiers journaux, et inviter la fonctionnalité à extraire les données des fichiers journaux. Le compte utilisateur qui exécute le programme Q Capture a besoin de droits d'accès spéciaux pour administrer la fonctionnalité LogMiner.

Configuration de la fonctionnalité Oracle LogMiner

Pour configurer la fonctionnalité Oracle LogMiner pour la réplication Q ou la publication d'événements, vous devez créer un espace table pour les tables redo de LogMiner et créer un compte administrateur LogMiner avec les droits d'accès appropriés.

A propos de cette tâche

Le LogMiner présente les données modifiées à partir du journal redo dans une pseudo-vue relationnelle appelée V\$LOGMNR_CONTENTS. Par défaut, la fonctionnalité utilise l'espace table SYSTEM pour la vue V\$LOGMNR_CONTENTS. Oracle recommande qu'un espace table différent soit créé pour les opérations LogMiner.

Pour permettre au programme Q Capture d'accéder aux vues LogMiner, vous devez créer des droits d'accès spécifiques. Les droits d'accès doivent également être accordés pour permettre au LogMiner d'accéder aux données dans les vues et tables système Oracle détenues par l'utilisateur SYS.

Des exemples de script SQL sont fournis dans le répertoire sqllib/samples/repl/oracle que vous pouvez modifier pour indiquer un espace table pour LogMiner et pour définir les droits d'accès adéquats. Vous pouvez également utiliser la procédure suivante pour accomplir ces tâches.

Recommandation : Alors qu'Oracle prend en charge les configurations dans lesquelles LogMiner exécute une base de données différente de l'instance de base de données qui génère les journaux redo et d'archivage, IBM ne recommande pas cette configuration pour la réplication Q. le programme Q appelle LogMiner avec l'option CONTINUOUS_MINE. Cette option invite LogMiner à identifier de manière dynamique et à ajouter les fichiers journaux dont il a besoin. Si vous exécutez LogMiner sur une base de données d'analyse distante, vous devrez manuellement rendre actifs les fichiers journaux et d'archive disponibles pour LogMiner.

Procédure

1. Créez un espace table pour la fonctionnalité LogMiner dans la base de données source Oracle. Procédez comme suit :
 - a. Emettez l'instruction SQL CREATE TABLESPACE SQL. Vous pouvez modifier l'exemple UNIX suivant pour qu'il corresponde à votre environnement :

```
CREATE TABLESPACE logmnrts$
DATAFILE
'/dbms/oracle10/product/10.1.0/oradata/ORA10SID/logmnr01.dbf'
SIZE 4M AUTOEXTEND ON
NOLOGGING EXTENT MANAGEMENT LOCAL;
```

- b. Affectez le nouvel espace table à la fonctionnalité LogMiner. L'instruction suivante affecte l'espace table logmnrts\$ qui a été créé au cours de l'étape a :

```
EXECUTE SYS.DBMS_LOGMNR_D.SET_TABLESPACE('logmnrts');
```

2. Créez un compte utilisateur LogMiner pour la réplication Q dans la source Oracle. Les exemples d'instructions suivantes créent l'asn utilisateur et accordent à l'utilisateur les droits d'accès appropriés ("asn" n'est pas un nom obligatoire pour le compte utilisateur, le nom peut être modifié afin de correspondre à vos besoins) :

```
créez un asn utilisateur identifié par asn
utilisateurs d'espace table par défaut
temp espace table temporaire
déverrouillage du compte par défaut du profil ;
accorder une connexion, ressource à asn ;
accorder une session créer à asn ;
modifier le quota asn utilisateur illimité sur logmnrts$ ;
accorder sélectionner toute transaction sur asn ;
accorder execute_catalog_role sur asn ;
accorder sélectionner toute table sur asn ;
accorder sélectionner on sys.v_$database sur asn ;
accorder sélectionner on sys.v_$logmnr_contents sur asn ;
accorder sélectionner select on sys.v_$logmnr_dictionary sur asn ;
accorder sélectionner on sys.v_$logmnr_logfile sur asn ;
accorder sélectionner on sys.v_$logmnr_logs sur asn ;
accorder sélectionner on sys.v_$logmnr_parameters sur asn ;
accorder sélectionner on sys.v_$logmnr_session sur asn ;
accorder sélectionner on sys.v_$logmnr_transaction sur asn ;
accorder sélectionner on sys.v_$log sur asn ;
accorder sélectionner on sys.v_$logfile sur asn ;
accorder sélectionner on sys.v_$archived_log sur asn ;
```

Création d'un fichier de configuration ASNCLP

Le programme de ligne de commande ASNCLP utilise un fichier de configuration pour la connexion aux sources de données Oracle. Le fichier de configuration est un fichier de texte normal.

A propos de cette tâche

Vous pouvez définir les informations de connexion pour plusieurs sources de données dans le fichier de configuration. Vous pouvez sauvegarder le fichier de configuration à l'emplacement de votre choix. La commande **SET SERVER** comporte un paramètre permettant d'indiquer au programme ASNCLP l'emplacement de votre fichier de configuration.

Procédure

Pour créer un sauvegarder ASNCLP :

1. Dans un fichier texte, ajoutez un nom unique entre crochets afin d'identifier la source de données. Le nom doit comporter huit caractère ou moins. Par exemple, [Oracle1].
2. Saisissez les paramètres TYPE, DATA SOURCE, HOST, PORT dans votre fichier de configuration et remplacez les exemples de valeur par vos propres valeurs. L'exemple suivant donne des valeurs pour une source Oracle :

```
[Oracle1]
Type=ORACLE
Data source=ORADB
Host=9.30.155.156
Port=1521
```

3. Enregistrez le fichier à l'emplacement de votre choix. Lorsque vous utilisez la commande **SET SERVER**, vous devez entrer le chemin d'accès complet au fichier. Voici un exemple de commande SET SERVER définissant un fichier de configuration ASNCLP :

```
SET SERVER capture TO CONFIG SERVER ORADB
FILE "c:\ora_setup\asnservers.ini"
ID myuserid PASSWORD "mypassword";
```

Création de tables de contrôle Q Capture pour une source Oracle

Avant de pouvoir publier ou répliquer des données à partir d'une source Oracle, vous devez créer des *tables de contrôle* pour un programme Q Capture dans la base de données Oracle source. Les tables de contrôle stockent des informations sur les abonnements Q, les publications, les files d'attente de messages, les paramètres opérationnels et les préférences utilisateurs.

Avant de commencer

- Le programme de ligne de commande ASNCLP ou le Centre de réplication doit être capable de se connecter à la base de données Oracle source sur laquelle vous souhaitez créer les tables de contrôle Q Capture. Pour permettre les connexions avec ASNCLP, créez un fichier de configuration ASNCLP.
- Vous devez posséder les noms des objets WebSphere MQ suivants :
 - Un gestionnaire de files d'attente avec lequel fonctionne le programme Q Capture
 - Une file d'attente locale et permanente pour servir de file d'attente d'administration
 - Une file d'attente locale et permanent pour servir de file d'attente de redémarrage.

Conseil : Utilisez la commande **VALIDATE WEBSPPHERE MQ ENVIRONMENT FOR** dans le ASNCLP, ou la commande **validate controls** dans le Centre de réplication pour vous assurer que les gestionnaires de files d'attente et les files d'attente que vous avez spécifiés pour les tables de contrôle existent et possèdent des propriétés correctes. Pour que cette commande **VALIDATE** fonctionne, le programme Q Capture et les objets WebSphere MQ doivent être situés sur le même serveur que la base de données Oracle.

A propos de cette tâche

Chaque instance du programme Q Capture possède son propre ensemble de tables de contrôle, identifiées par le schéma Q Capture. Par exemple, la table de contrôle stockant les paramètres opérationnels d'un programme Q Capture avec un schéma de ASN1 serait nommée ASN1.IBMQREP_CAPPARMS.

Les tables de contrôle Q Capture sont créées de manière native dans la base de données Oracle, par défaut dans un espace table unique. Vous pouvez personnaliser l'emplacement de la création de chaque table de contrôle et vous pouvez également spécifier les espaces table existants ou en créer de nouveaux.

Procédure

Pour créer des tables de contrôle Q Capture dans une source Oracle, utilisez l'une des méthodes suivantes :

Méthode	Description
<p>Programme de ligne de commande ASNCLP</p>	<p>Utilisez la commande CREATE CONTROL TABLES FOR. Par exemple, les commandes suivantes définissent l'environnement et créent des tables de contrôle dans la base de données Oracle SAMPLE avec un schéma Q Capture de ASN1 :</p> <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE CONFIG SERVER SAMPLE FILE asnservers.ini ID id1 PASSWORD pwd1; SET QMANAGER "QM1" FOR CAPTURE SCHEMA; SET CAPTURE SCHEMA SOURCE ASN1;</pre> <pre>CREATE CONTROL TABLES FOR CAPTURE SERVER USING RESTARTQ "ASN1.QM1.RESTARTQ" ADMINQ "ASN1.QM1.ADMINQ" MEMORY LIMIT 64 MONITOR INTERVAL 600000;</pre> <p>La commande CREATE spécifie une file d'attente de redémarrage et une file d'attente d'administration, elle double la taille par défaut de mémoire disponible pour construire des transactions à 64 Mo, et réduit à 600000 millisecondes (une minute) la fréquence par défaut d'enregistrement des informations sur la performance.</p>
<p>Centre de réplication</p>	<p>Lorsque vous créez des tables de contrôle Q Capture sur une source Oracle ou lorsque vous ajoutez un serveur Q Capture existant au centre de réplication, utilisez la fenêtre Indiquer un serveur Oracle.</p> <p>Cliquez-droit sur le dossier Serveurs Q Capture puis cliquez sur Créer des tables de contrôle Q Capture. Cliquez sur Ajouter un serveur Oracle et indiquez la base de données source Oracle dans la fenêtre Indiquer un serveur Oracle :</p> <p>Nom de base de données Le service Oracle, ou identificateur du système de la base de données qui contient les tables de contrôle Q Capture</p> <p>Alias de base de données Un alias TNS de 8 caractères pour la base de données qui contient les tables de contrôle Q Capture</p> <p>Nom de système Le nom d'hôte ou l'adresse IP du système dans lequel s'exécute la base de données</p> <p>Numéro de port Le numéro de port dans lequel la base de données recherche des demandes de connexion entrantes</p> <p>ID utilisateur Un ID utilisateur pour se connecter au serveur source Oracle</p> <p>Mot de passe Un mot de passe pour se connecter au serveur source Oracle</p>

Création d'abonnements Q pour les sources Oracle

Un abonnement Q mappe la table Oracle qui contient vos données source dans une copie de cette table dans la cible DB2 ou fédérée. Lorsque vous créez un abonnement Q, vous définissez une mappe de files d'attente de réplication, des options de table cible et d'autres préférences. Vous créez un abonnement Q pour chaque table que vous souhaitez répliquer.

Avant de commencer

- Planifiez la manière dont vous voulez regrouper les mappes de files de réplication et les abonnements Q.
- Créez les tables de contrôle du programme Q Capture dans le serveur source Oracle qui contient la table source pour l'abonnement Q.
- Créez les tables de contrôle pour le programme Q Apply dans le serveur contenant la cible pour l'abonnement Q.
- Définissez les files d'attente pour la réplication et leurs attributs en créant une mappe de file de réplication. (Vous pouvez effectuer cette tâche avant de créer un abonnement Q ou pendant que vous créez un abonnement Q.)
- Préparez la procédure stockée si vous souhaitez que le programme Q Apply envoie des modifications source plutôt dans une procédure stockée que dans une table cible.

Si vous choisissez de charger automatiquement les tables cible avec des données d'une source Oracle, vous devez utiliser une base de données fédérée. Les fonctions de fédération pour les bases de données Oracle et autres, sont intégrées dans InfoSphere Replication Server. Les fonctions fédérées permettent au programme Q Apply de se connecter et d'extraire les données de la source Oracle dans les tables cible. Vous devez créer les objets suivants dans la base de données fédérée qui pointe vers les tables source Oracle :

- Encapsuleur Oracle
- Définition de serveur distant
- Mappage utilisateur

Restrictions

- Les abonnements Q multidirectionnels ne sont pas pris en charge pour les sources Oracle.
- Si vous utilisez une condition de recherche pour filtrer les modifications à répliquer, le prédicat doit utiliser une syntaxe Oracle valide.
- Les opérations de table tronquées dans la base de données source Oracle ne sont pas répliquées.

Procédure

Pour créer des abonnements Q pour les sources Oracle :

Utilisez l'une des méthodes suivantes :

Méthode	Description
<p>Programme de ligne de commande ASNCLP</p>	<p>Utilisez la commande CREATE QSUB. Par exemple, les commandes suivantes définissent l'environnement et créent un abonnement Q avec les caractéristiques suivantes :</p> <ul style="list-style-type: none"> • Le serveur source Oracle est LONDON et les informations de connexion ASNCLP sont stockées dans le fichier asnservers.ini sous l'entrée LONDON. • Le serveur cible DB2 est DALLAS. • La mappe de files d'attente de réplication est LONDON_ASN_TO_DALLAS_ASN • La table source est SALES et la table cible est TGTSALES. • Le nom d'abonnement Q est SALES_REPORTING. • La table est automatiquement chargée à l'aide de l'option LOAD FROM CURSOR (HAS LOAD PHASE I et LOAD TYPE 1). <pre> ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO CONFIG SERVER LONDON FILE asnservers.ini ID MYUSERID PASSWORD MYPASS; SET SERVER TARGET TO DB DALLAS; SET CAPTURE SCHEMA SOURCE ASN; SET APPLY SCHEMA ASN; SET QMANAGER QM1 FOR CAPTURE SCHEMA; SET QMANAGER QM2 FOR APPLY SCHEMA; CREATE QSUB (SUBNAME SALES_REPORTING REPLQMAP LONDON_ASN_TO_DALLAS_ASN SALES OPTIONS HAS LOAD PHASE I TARGET NAME TGTSALES TYPE USERTABLE NEW NICKNAME RMT SERVERNAME RMTLONDN NICKSALES LOAD TYPE 1); </pre>
<p>Centre de réplication</p>	<p>Utilisez l'assistant Création d'abonnements Q. Pour ouvrir l'assistant, développez le schéma Q Capture ou Q Apply approprié dans l'arborescence des objets, cliquez avec le bouton droit de la souris sur le dossier Abonnements Q , et sélectionnez Créer.</p> <p>Page Serveurs Indiquez le serveur Oracle sur lequel le programme Q Capture s'exécute en tant que serveur source.</p> <p>Page des lignes et des colonnes la colonne Type de données affiche le type de données Oracle de la colonne. Le programme Q Capture convertit les types de données Oracle pris en charge en type de données natifs de DB2. Vous pouvez afficher le mappage du type de données dans la fenêtre Mappage de colonne. Pour plus de détails, voir les types de données Oracle.</p> <p>Page de chargement des tables cible Si vous voulez effectuer un chargement automatique des tables cible, sélectionnez le bouton de radio Automatique et l'option Toujours utiliser LOAD FROM CURSOR du bouton de radio Fonctionnalité LOAD.</p> <p>Utilisez les commandes de la zone Pseudonyme pour inviter le centre de réplication à créer un nouveau pseudonyme ou à indiquer un pseudonyme existant pour le processus de chargement.</p>

Options de chargement pour la réplication Q à partir d'une source Oracle

Lorsque la table source d'un abonnement Q se trouve dans une base de données Oracle, vous pouvez choisir entre les options de chargement automatique, manuel et pas de chargement pour la table cible.

Le tableau suivant décrit chaque option.

Tableau 15. Options de chargement pour les abonnements Q avec des sources Oracle

Option	Description
Automatique	<p>Pour charger automatiquement des tables cibles, utilisez la fonction de base de données fédérée dans InfoSphere Replication Server pour inviter le programme Q Apply à extraire les données des tables source Oracle pour les envoyer aux tables DB2 source.</p> <p>Q Apply s'appuie sur l'option LOAD FROM CURSOR de l'utilitaire DB2 LOAD. L'utilitaire LOAD extrait les données Oracle à l'aide de pseudonymes dans la base de données fédérée.</p> <p>La configuration du chargement diffère selon la cible DB2 :</p> <p>z/OS</p> <p>Pour les tables cibles de DB2 pour z/OS, vous devez vous appuyer sur une base de données fédérée sous Linux, UNIX ou Windows, de sorte que l'utilitaire LOAD puisse charger les tables cibles dans DB2 pour z/OS. Effectuez la mise à jour de la configuration du gestionnaire de base de données à l'aide de FEDERATED YES, créez une base de données fédérée et fournissez à la base de données des informations sur votre base de données et vos tables Oracle.</p> <p>Remarque : Le serveur hébergeant la base de données fédérée doit posséder le client Oracle en état de marche pour permettre les connexions à la base de données Oracle.</p> <p>Pour configurer le processus de chargement, vous devez télécharger deux scripts exemples depuis la collection d'exemples de réplication d'IBM developerWorks :</p> <p>001_federated_setup.db2 Ce script crée une base de données fédérée, un encapsuleur Oracle, une définition de serveur fédéré, un mappage utilisateur et un pseudonyme pour la table source Oracle.</p> <p>002_zos_setup.db2 Ce script permet l'accès cible z/OS au pseudonyme fédéré créé dans 001_federated_setup.db2.</p> <p>Remarque : Créez des abonnements Q pour les tables source Oracle avant d'utiliser ces exemples. La base de données et le pseudonyme fédérés doivent correspondre à la table source et à la base de données Oracle, de sorte que Q Apply utilise LOAD FROM CURSOR.</p> <p>Linux UNIX Windows</p> <p>Pour les cibles réparties, vous devez activer la fonction de base de données fédérée dans la base de données cible et fournir au serveur fédéré les informations concernant votre base de données et vos tables Oracle.</p> <p>Pour pouvoir créer des abonnements Q mappés entre les tables sources Oracle et les tables cibles DB2, vous devez créer les objets suivants sur le serveur fédéré (ce qui active l'accès à Oracle) :</p> <ul style="list-style-type: none"> • Encapsuleur Oracle • Définition de serveur distant • Mappage utilisateur <p>Lorsque vous créez des abonnements Q qui utilisent LOAD FROM CURSOR, vous indiquez les objets fédérés à utiliser. Vous pouvez utiliser le Centre de réplication ou le programme de ligne de commande ASNCLP pour créer les pseudonymes qui pointent vers les tables source Oracle, ou vous pouvez utiliser les pseudonymes existants.</p>
Manuel	<p>Le chargement manuel dans lequel vous chargez les tables cible avec la fonctionnalité de votre choix, puis indiquez aux programmes de réplication quand le chargement est terminé, est pris en charge pour les sources Oracle. par exemple, vous pouvez utiliser IBM Migration Toolkit pour charger les tables cible avec des données de la source Oracle, puis utiliser l'ASNCLP ou le centre de réplication pour indiquer quand le chargement est terminé.</p>

Tableau 15. Options de chargement pour les abonnements Q avec des sources Oracle (suite)

Option	Description
Aucun chargement	<p>Lorsque vous spécifiez l'option aucun chargement pour un abonnement Q, le programme Q Apply commence à appliquer des transactions sur une table cible dès que l'abonnement devient actif.</p> <p>Si vous choisissez cette option, assurez-vous que les valeurs de la clé primaire ou de l'index à entrées uniques de la table source figurent également dans la clé primaire ou l'index à entrées uniques de la table cible.</p>

Démarrage d'un programme Q Capture pour des sources Oracle

Démarrez un programme Q Capture en commençant à capturer les transactions ou modifications au niveau de la ligne à partir d'Oracle pour les abonnements Q ou publications actifs ou nouveaux.

Avant de commencer

- Créez un gestionnaire de files d'attente WebSphere MQ, des files d'attente et d'autres objets requis.
- Assurez-vous que vous avez les droits requis pour les objets de réplication Q et les objets de publication d'événements ainsi que pour les objets WebSphere MQ.
- Créez des tables de contrôle sur la base de données source Oracle pour le schéma Q Capture approprié.
- Configurez la base de données source Oracle pour qu'elle fonctionne avec le programme Q Capture.

A propos de cette tâche

Lors du démarrage initial d'un programme Q Capture sans spécification du mode de démarrage, le programme utilise le mode de démarrage par défaut, warmsi. Par ce mode, le programme essaie de lire le journal à l'emplacement où il s'était arrêté. Dans la mesure où le programme démarre pour la première fois, il passe en mode de démarrage à froid et démarre le traitement des abonnements Q ou des publications qui sont à l'état N (nouveau) ou A (actif). Tous les abonnements Q ou publications qui sont à l'état I (inactif) doivent être activés pour que le programme commence à capturer les modifications.

Vous pouvez démarrer un programme Q Capture même si aucun abonnement Q ou publication n'est à l'état A (actif). Lorsque vous activez les abonnements Q ou les publications, le programme Q Capture commence à capturer les modifications pour ces abonnements Q ou publications.

Lorsque vous démarrez un programme Q Capture, vous pouvez définir les valeurs des paramètres de démarrage et le programme utilisera les nouvelles valeurs jusqu'à ce que vous faisiez l'une des actions suivantes :

- Changer les valeurs des paramètres pendant que le programme est exécuté.
- Arrêter et redémarrer le programme, ce qui l'invite à lire la table IBMQREP_CAPPARMS et utiliser les valeurs sauvegardées ici.

Procédure

Pour démarrer un programme Q Capture, appliquez l'une des méthodes suivantes :

Méthode	Description
commande asnoqcap	<p>Utilisez la commande asnoqcap pour démarrer un programme Q Capture pour les sources Oracle et indiquez les paramètres de démarrage. Par exemple :</p> <pre>asnoqcap capture_server=nom_serveur capture_schema=schéma paramètres</pre> <p>Où <i>nom_serveur</i> est le nom de la base de données contenant les tables de contrôle Q Capture, <i>schéma</i> identifie le programme Q Capture que vous souhaitez démarrer et <i>paramètres</i> correspond à un ou plusieurs des paramètres que vous pouvez définir au démarrage.</p>
<div style="background-color: #800000; color: white; padding: 2px; text-align: center; font-weight: bold;">Windows</div> Services Windows	Vous pouvez créer un service de réplication sous les systèmes d'exploitation Windows pour démarrer le programme Q Capture automatiquement quand le système est lancé.

Vous pouvez vérifier si un programme Q Capture a démarré en appliquant une des méthodes suivantes :

- Emettez la commande **asnoqcmd status**.
- Examinez le fichier journal de diagnostic Q Capture (*serveur_capture.schéma_capture.QCAP.log*) pour un message qui indique que le programme capture les modifications.
- Recherchez dans la table IBMQREP_CAPTRACE un message indiquant que le programme capture les modifications.

Comment Q Capture gère les modifications des tables source Oracle

Lorsque vous modifiez une table source Oracle, le programme Q Capture continue de répliquer les données tant que les définitions de colonnes existantes ne sont pas modifiées. Les modifications de table ne sont pas automatiquement répliquées sur la cible.

Le tableau suivant décrit comment Q gère différents types de DDL et ce dont vous avez besoin pour tout abonnement Q affecté :

Tableau 16. Gestion par Q Capture des modifications de langage de définition de données apportées aux tables source et ce que vous devez faire

Opération de langage de définition de données	Comment elle est traitée	Action à accomplir
ALTER TABLE ADD (COLUMN)	Q Capture laisse l'abonnement Q actif mais ne réplique pas la colonne ajoutée jusqu'à la réception d'un signal ADDCOL.	Utilisez la commande ASNCLP ALTER ADD COLUMN ou insérez manuellement un signal ADDCOL pour indiquer que vous voulez répliquer la nouvelle colonne.
DROP TABLE	Q Capture laisse l'abonnement Q actif, mais il n'y aura plus d'enregistrements de journaux à lire pour la table source.	Arrêtez l'abonnement Q, puis supprimez l'abonnement Q.

Tableau 16. Gestion par Q Capture des modifications de langage de définition de données apportées aux tables source et ce que vous devez faire (suite)

Opération de langage de définition de données	Comment elle est traitée	Action à accomplir
<p>Langage de définition de données qui altère la structure d'une table</p> <p>Exemples :</p> <ul style="list-style-type: none"> • DROP COLUMN • SET UNUSED • MODIFY COLUMN • RENAME COLUMN 	<p>L'abonnement Q pour toute table modifiée est arrêté. Q Capture continue de fonctionner.</p>	<ol style="list-style-type: none"> 1. Supprimez et recréez l'abonnement Q pour synchroniser les définitions de table source et cible dans les tables de contrôle Q Capture et Q Apply. 2. Redémarrez l'abonnement Q pour déclencher un nouveau chargement de la table cible.
<p>Langage de définition de données qui n'altère pas la structure d'une table</p> <p>Exemples :</p> <ul style="list-style-type: none"> • CREATE INDEX • ADD/MODIFY CONSTRAINT • ADD DEFAULT VALUE 	<p>Q Capture laisse l'abonnement Q actif.</p>	<p>Vérifiez que les contraintes d'unicité, les clés principales et les contraintes d'intégrité référentielle correspondent entre les tables source et cible. Si vous modifiez l'une de ces propriétés sur la table source, effectuez la modification correspondante sur la table cible pour éviter un comportement imprévu.</p>

Types de données Oracle

Le programme Q Capture convertit les types de données Oracle pris en charge en type de données natifs de DB2. Certains types de données Oracle ne sont pas pris en charge par la réplication Q ou la publication d'événements.

Types de données Oracle pris en charge

Les types de données suivants sont pris en charge par la réplication Q et la publication d'événements :

- CHAR
- NCHAR
- VARCHAR2
- NVARCHAR2
- NUMBER
- DATE
- TIMESTAMP
- RAW
- CLOB
- NCLOB
- BLOB
- BINARY_FLOAT

- BINARY_DOUBLE

La prise en charge des types de données LOB n'est disponible que pour les journaux redo qui sont générés par une base de données avec une compatibilité définie sur la valeur 9.2.0.0 ou supérieure.

Vous ne pouvez pas répliquer de colonnes LONG ou LONG RAW. Si vous convertissez ces colonnes en colonnes LOB, vous pouvez répliquer les données.

Mappage du type de données

Le programme Q Capture convertit les types de données Oracle pris en charge en type de données natifs de DB2. Le programme Q Capture convertit les types de données conformément au tableau ci-dessous.

Tableau 17. Mappage de type de Oracle en DB2

Type de données Oracle	Longueur inférieure Oracle	Longueur supérieure Oracle	Echelle inférieure Oracle	Echelle supérieure Oracle	Données binaires Oracle	Type de données DB2	Longueur DB2	Echelle DB2	Données binaires DB2
BLOB	0	0	0	0	-	BLOB	2147483647	0	Y
CHAR	1	254	0	0	-	CHAR	0	0	N
CHAR	255	2000	0	0	-	VARCHAR	0	0	N
CLOB	0	0	0	0	-	CLOB	2147483647	0	N
DATE	0	0	0	0	-	TIMESTAMP	0	0	N
FLOAT	1	126	0	0	-	DOUBLE	0	0	N
LONG	0	0	0	0	-	CLOB	2147483647	0	N
LONG RAW	0	0	0	0	-	BLOB	2147483647	0	Y
NUMBER	1	38	-84	127	-	DOUBLE	0	0	N
NUMBER	1	31	0	31	-	DECIMAL	0	0	N
NUMBER	1	4	0	0	-	SMALLINT	0	0	N
NUMBER	5	9	0	0	-	INTEGER	0	0	N
NUMBER	-	10	0	0	-	DECIMAL	0	0	N
RAW	1	2000	0	0	-	VARCHAR	0	0	Y
ROWID	0	0	0	NULL	-	CHAR	18	0	N
TIMESTAMP	-	-	-	-	-	TIMESTAMP	10	-	-
VARCHAR2	1	4000	0	0	-	VARCHAR	0	0	N

Limitations du type de données

Certaines valeurs NUMBER Oracle peuvent être arrondies par la réplication.

Les types de données Oracle sont convertis en types de données DB2 lorsqu'ils sont répliqués. En raison des différences entre les types de données mappées, les NUMBERS Oracle peuvent être arrondis en fonction de leur longueur et de leur échelle.

Les colonnes NUMBER Oracle avec les attributs suivants peuvent être arrondies :

- Les colonnes NUMBER définies avec une longueur de 31 à 38 peuvent être arrondies.
- Les colonnes NUMBER définies avec une longueur de 1 à 31 et une échelle de -84 à 0, ou une échelle de 31 à 127 peuvent être arrondies.

Remarque : Lorsque la longueur n'est pas spécifiée pour un NUMBER, celui-ci est traité comme ayant une longueur de 38 et une échelle 0. Le type de données DOUBLE DB2 a un maximum de 17 chiffres significatifs. Cela signifie que si une valeur a plus de 17 chiffres significatifs, la valeur sera arrondie. par exemple, si votre valeur NUMBER Oracle est 9876543210987654321, votre valeur DOUBLE DB2 sera

9.8765432109876543E 16, qui arrondit les deux derniers chiffres de la valeur Oracle.

Pour éviter l'arrondi, spécifiez la longueur (et l'échelle si nécessaire) dans NUMBER lorsqu'une colonne est déclarée.

L'utilisation des colonnes TIMESTAMP Oracle en tant que clés pour la réplication ou la publication nécessite le groupe de correctifs 10.2.0.5.

Les colonnes TIMESTAMP pour les tables source Oracle ne peuvent être utilisées comme clés pour la réplication ou la publication (IS_KEY=Y) que si vous installez le groupe de correctifs Oracle 10.2.0.5. Sans la mise à jour, la fonctionnalité LogMiner ne renvoie pas la valeur inférieure à la seconde, et lorsque vous répliquez des valeurs d'horodatage, la précision est en secondes plutôt qu'en dixièmes de secondes.

Chapitre 10. Manipulation des données à l'aide des procédures stockées en tant que cibles

Lorsque vous créez un abonnement Q pour une réplication unidirectionnelle, vous pouvez spécifier que les modifications soient répliquées de la table source vers une procédure stockée plutôt que directement vers une table cible. Vous pouvez utiliser la procédure stockée pour manipuler des données capturées depuis une table source avant qu'elles ne soient appliquées à la table cible.

Procédures stockées pour la manipulation des données source pour la réplication Q

Généralement, dans les abonnements Q, les données provenant d'une table source sont mappées vers une table cible ; cependant, pour des abonnements Q dans une réplication unidirectionnelle, vous pouvez avoir à la place le programme Q Apply qui appelle une procédure stockée et transmette les données source en tant que paramètres d'entrée vers la procédure stockée.

Les colonnes source mappent vers des paramètres dans la procédure stockée au lieu de mapper vers des colonnes cible. En mappant directement des colonnes source vers des paramètres dans une procédure stockée, vous évitez le besoin d'analyser les données entrantes et vous obtenez un modèle de programmation simple et propre.

Lorsque vous spécifiez que vous voulez que la cible d'un abonnement Q soit une procédure stockée, vous avez toujours toutes les options dont vous disposez pour un abonnement Q avec une table cible. Par exemple, avec une procédure stockée, vous faites toujours le choix d'options d'erreur. La procédure stockée peut faire référence à une table DB2 ou à un alias dans une base de données fédérée DB2.

La procédure stockée renvoie un code retour au programme Q Apply qui indique si les données sont appliquées à la table cible. Le développeur qui crée une procédure stockée doit s'assurer que la procédure stockée renvoie un code retour SQL approprié au programme Q Apply. La procédure stockée se charge transmettre les données source vers leur destination finale.

Version 10.1 et ultérieure : Le paramètre de procédure stockée obligatoire `src_commit_1sn` doit indiquer un type de données VARCHAR(16) dans les conditions suivantes :

- La base de données source est DB2 10.1 pour Linux, UNIX et Windows, ou une version ultérieure
- Le programme Q Capture est la version 10.1 ou ultérieure et la valeur du paramètre **compatibility** est 1001 ou supérieur

Si la version de la base de données source et du programme Q Capture est V10.1, et que la valeur du paramètre **compatibility** est inférieure à 1001, la valeur du paramètre `src_commit_1sn` peut être CHAR(10), car Q Capture continue d'utiliser des numéros d'ordre du journal de 10 octets.

Exemple

L'exemple suivant décrit une signature d'une procédure stockée qui accepte des valeurs provenant de cinq colonnes de la table source, deux d'entre elles étant des colonnes de clé.

```
CREATE PROCEDURE storedprocedure_name(  
    INOUT operation integer,  
    IN suppression_ind VARCHAR(size),  
    IN src_commit_lsn CHAR(10),  
    IN src_trans_time TIMESTAMP,  
    XParm1,  
    Parm1,  
    XParm2,  
    Parm2,  
    Parm3,  
    Parm4,  
    Parm5  
)
```

Cet exemple illustre les quatre paramètres obligatoires :

operation

Ce paramètre INOUT identifie le type d'opération.

suppression_ind

Ce paramètre IN identifie les paramètres ayant été supprimés.

src_commit_lsn

Ce paramètre IN identifie le numéro d'ordre du journal du moment où le serveur source a émis COMMIT pour la transaction.

src_trans_time

Ce paramètre IN identifie l'horodatage du moment où le serveur source a émis COMMIT pour la transaction.

La signature contient également les valeurs avant et après des colonnes de clé et uniquement les valeurs après des colonnes de non clé. Les paramètres suivants acceptent les valeurs des colonnes source :

- Les paramètres Parm1 et Parm2 mappent vers des colonnes de clé de la table source.
- Les paramètres XParm1 et XParm2 acceptent les valeurs avant des colonnes de clé de la table source.
- Les paramètres Parm3, Parm4 et Parm5 mappent vers des colonnes de non clé de la table source.

Voir les exemples de programmes pour obtenir des exemples de procédures stockées pour la réplication Q écrites en langage C et SQL.

La procédure stockée ne doit pas effectuer les fonctions COMMIT ou ROLLBACK car le programme Q Apply valide ou annule les transactions.

Le programme Q Apply gère les données LOB pour les procédures stockées de manière identique à la gestion des données LOB pour les tables cible. Pour les abonnements Q avec des procédures stockées, les modifications d'objet LOB sont répliquées par leur envoi en plusieurs messages, en fonction de la taille de l'objet LOB et de celle du message que vous avez autorisé le programme Q Capture à envoyer. L'appel de la procédure stockée qui appelle le premier message d'objet LOB pour une modification d'objet LOB unique est l'opération sur la ligne d'origine. Toutes les données d'objet LOB restantes pour la modification d'objet

LOB unique sont envoyées par des appels supplémentaires vers la procédure stockée. Le programme Q Apply transforme l'opération en une mise à jour. L'index `suppression_ind` marque tous les paramètres comme supprimés, sauf le paramètre qui mappe vers la colonne d'objet LOB de la table source. La procédure stockée est appelée à chaque fois qu'un message d'objet LOB se trouve dans la file d'attente de réception.

Si une valeur d'objet LOB est suffisamment faible pour être envoyée dans un seul message, l'opération de mise à jour spéciale avec ajout (valeur d'opération 34) n'est pas nécessaire. Le programme d'écriture de la procédure stockée doit procéder à une écriture logique pour gérer chaque code d'opération ainsi que la suppression de colonne ; le programme Q Apply a une logique totalement différente pour la gestion des valeurs d'objet LOB.

Si une valeur d'objet LOB est trop grande pour être envoyée en un seul message (lorsque la taille dépasse la taille maximale du message) et que, par conséquent, elle est divisée en différents messages, le premier message est géré différemment des messages suivants. Le programme Q Apply ajoute les parties d'une valeur d'objet LOB non envoyées dans le premier message en utilisant l'opération de concaténation `||`. Un code d'opération spécial est transmis pour indiquer cette forme spéciale de mise à jour.

Exemple d'insertion LOB

Dans une procédure stockée disposant de deux colonnes LOB, C1 et C2, il se produirait ce qui suit dans une insertion :

SQL	Valeur de l'opération
INSERT C1 = first portion C2 = some dummy value	16
UPDATE C1 = C1 next portion	34
UPDATE C1 = C1 next portion	34
UPDATE C2 = first portion	32
UPDATE C2 = C2 next portion	34
UPDATE C2 = C2 next portion	34

Écriture de procédures stockées pour manipuler des données source pour la réplication Q

Vous pouvez écrire une procédure stockée afin de l'utiliser comme cible dans un abonnement Q. Vous devez écrire la procédure stockée avant de pouvoir la déclarer comme cible dans un abonnement Q.

Restrictions

- La procédure stockée ne doit pas effectuer les fonctions COMMIT ou ROLLBACK.
- Vous ne pouvez pas envoyer la valeur avant à partir d'une colonne de non clé vers un paramètre dans une procédure stockée.
- Puisque le programme Q Apply ne possède des informations que sur la procédure stockée et non pas sur la table vers laquelle la procédure stockée est

susceptible de transmettre les données manipulées, les abonnements Q ayant des procédures stockées qui leur sont associées ne peuvent pas maintenir une intégrité référentielle.

- Si la procédure stockée gère une table cible et si vous souhaitez que le programme Q Apply résolve les conflits, alors ne créez pas d'index secondaires à entrées uniques sur la table cible.

A propos de cette tâche

La procédure stockée comprend quatre paramètres obligatoires (un qui identifie l'opération et trois qui identifient la transaction) et des paramètres supplémentaires qui mappent vers des colonnes source. Vous pouvez écrire une procédure stockée qui fait référence à une table DB2 ou à un pseudonyme dans une base de données fédérée DB2.

Procédure

Pour écrire une procédure stockée pour qu'elle soit une cible dans un abonnement Q et pour la configurer :

1. Ecrivez une procédure stockée contenant quatre paramètres obligatoires et des paramètres supplémentaires qui mappent vers des colonnes source. Les paramètres de la procédure stockée sont positionnels. Le premier paramètre doit avoir le mode de paramètre INOUT. Tous les autres paramètres doivent avoir le mode de paramètre IN. Reportez-vous aux rubriques suivantes :
 - «Paramètre de procédure stockée qui identifie le type d'opération de la réplication Q», à la page 185
 - «Paramètre de procédure stockée qui indique si chaque colonne source a été supprimée», à la page 187
 - «Paramètres de procédure stockée qui identifient la transaction pour la réplication Q», à la page 188, comprenant les deux paramètres obligatoires suivants :
 - Le numéro d'ordre du journal du moment où le serveur source a émis l'instruction de validation pour la transaction
 - L'horodatage du moment où le serveur source a émis l'instruction de validation pour la transaction
 - «Paramètres de procédure stockée qui mappent vers des colonnes source pour la réplication Q», à la page 188

Conseil : voir les exemples de programmes pour obtenir des exemples de procédures stockées pour la réplication Q écrites en langage C et SQL.

2. Exécutez de manière appropriée l'instruction CREATE PROCEDURE de telle manière que le nom et les paramètres appropriés de la procédure stockée soient enregistrés avec la base de données DB2.

Conseils :

- Utilisez le style de paramètre GENERAL WITH NULLS lorsque vous déclarez la procédure stockée. Ce style accroît la portabilité de la procédure stockée sur l'ensemble de la famille DB2.
 - **Linux UNIX Windows** Assurez-vous que la procédure stockée existe dans le répertoire sqllib/function.
3. Compilez et liez la procédure stockée au niveau du serveur cible.
 - z/OS** Incluez le module d'interrogation de base de données dans la liste de modules Q Apply.

4. Assurez-vous que le programme Q Apply dispose des droits d'accès appropriés pour appeler la procédure stockée.
Après l'appel vers la procédure stockée, le programme Q Apply vérifie toujours le code SQL avant de vérifier le code retour pour valider le fait que la procédure stockée ait réussi ou non.
5. Si la procédure stockée gère une cible, assurez-vous que la cible existe.
6. Déclarez la procédure stockée comme cible dans un abonnement Q.

Paramètre de procédure stockée qui identifie le type d'opération de la réplication Q

Les attributs que vous spécifiez pour l'abonnement Q déterminent la manière dont le programme Q Apply gère les conditions inattendues dans la cible (dans le cas présent, dans la procédure stockée).

Déclarez le premier paramètre d'une procédure stockée comme paramètre INOUT que le programme Q Apply utilise pour transmettre le type d'opération (INSERT, UPDATE, DELETE ou KEY UPDATE) et que la procédure stockée utilise pour transmettre un code retour au programme Q Apply afin d'indiquer si l'opération s'est déroulée avec succès.

Conseil : Voir les exemples de programmes pour obtenir des exemples de paramètre `operation` dans les procédures stockées écrites en langage C et SQL. Notez la manière dont le paramètre `operation` est utilisé pour transmettre le code retour au programme Q Apply.

Le tableau 18 illustre les valeurs d'opération que le programme Q Apply transmet à la procédure stockée et ce que chaque valeur signifie.

Tableau 18. Codes retour SQL que le programme Q Apply transmet à la procédure stockée

Valeur de l'opération	Type d'opération
64	Suppression
16	Insertion
32	Mise à jour sur des colonnes de non clé
128	Mise à jour sur des colonnes de clé
34	Mise à jour et ajout

Définissez le paramètre INOUT `operation integer` sur le code SQL du programme Q Apply pour évaluer et voir si la modification a été appliquée avec succès sur la cible. Si la procédure stockée ne renvoie pas de code SQL au programme Q Apply, alors le programme Q Apply ne sait pas ce qui s'est passé au sein de la procédure stockée.

La procédure stockée ne doit pas effectuer les fonctions COMMIT ou ROLLBACK car le programme Q Apply valide ou annule les transactions.

Si d'autres applications apportent également des modifications à la table à laquelle la procédure stockée applique des modifications, il est alors possible que la procédure stockée soit confrontée à des conditions inattendues dans cette table. Par exemple, la procédure stockée peut essayer de mettre à jour une ligne déjà supprimée par une autre application. Vous disposez des choix suivants en ce qui

concerne la manière de résoudre des conditions inattendues qui peuvent se produire lorsque la procédure stockée tente d'insérer les données source manipulées dans la table cible :

- «Le programme Q Apply gère la condition inattendue»
- «La procédure stockée gère la condition inattendue»

Le programme Q Apply gère la condition inattendue

Si vous voulez que le programme Q Apply traite la condition inattendue dans la cible que la procédure stockée gère, écrivez alors la procédure stockée de telle manière qu'elle renvoie l'un des codes retour SQL répertoriés ci-dessous (tableau 19) au programme Q Apply et définissez l'abonnement Q pour le programme Q Apply pour forcer l'application des modifications dans la cible gérée par la procédure stockée. Le programme Q Apply force l'application de la modification de la source. Dans le cas de certains codes retour SQL, le programme Q Apply transforme l'opération de ligne, consigne l'exception dans la table IBMQREP_EXCEPTIONS et transmet l'opération de ligne transformée à la procédure stockée.

La procédure stockée gère la condition inattendue

Si vous ne souhaitez pas que le programme Q Apply gère les conditions inattendues dans la cible en forçant l'application des modifications apportés à la source dans la cible, vous pouvez générer une autre logique de traitement d'erreurs dans la procédure stockée. La procédure stockée doit toujours renvoyer un code SQL au programme Q Apply, mais elle peut renvoyer la valeur 0 (zéro) si bien que le programme Q Apply ne possède aucune information sur les conditions inattendues ou les échecs actuels au sein de la table. Une fois que la procédure stockée a transmis les codes retour SQL au programme Q Apply dans le paramètre `operation`, le programme Q Apply interprète chaque code SQL et démarre l'action appropriée (en fonction de la manière dont vous avez spécifié au programme Q Apply de traiter les conditions inattendues). Le programme Q Apply traite les codes retour de +100 et -803 en tant que conflits et tout autre code retour en tant qu'erreur. Le tableau 19 illustre les codes retour +100 et -803 que la procédure stockée génère, comment le programme Q Apply interprète ce type de code retour et quelle action le programme Q Apply entreprend en conséquence. Les informations contenues dans le tableau ci-dessous supposent que vous avez spécifié au programme Q Apply de forcer les modifications apportées à la cible.

Tableau 19. Codes retour SQL que la procédure stockée transmet au programme Q Apply et manière dont le programme Q Apply répond.

Code retour SQL	Type d'opération	Ce que le code retour signifie	Comment Q Apply réagit
0	Insertion	La ligne a été insérée avec succès.	Q Apply traite la ligne suivante.
0	Mise à jour	La ligne a été mise à jour avec succès.	Q Apply traite la ligne suivante.
0	Suppression	La ligne a été supprimée avec succès.	Q Apply traite la ligne suivante.
+100	Suppression	La ligne est introuvable dans la cible.	Q Apply ne relance pas l'appel.

Tableau 19. Codes retour SQL que la procédure stockée transmet au programme Q Apply et manière dont le programme Q Apply répond. (suite)

Code retour SQL	Type d'opération	Ce que le code retour signifie	Comment Q Apply réagit
+100	Mise à jour	La ligne est introuvable dans la cible.	Si vous avez spécifié à Q Apply de forcer les modifications, Q Apply modifie la mise à jour en une insertion et tente à nouveau l'appel.
-803	Insertion	La ligne existe déjà dans la cible	Si vous avez spécifié à Q Apply de forcer les modifications, Q Apply transforme l'insertion en une mise à jour et tente à nouveau l'appel.
-803	Mise à jour de la clé	La nouvelle clé existe déjà dans la cible.	Si vous avez spécifié à Q Apply de forcer les modifications, Q Apply transforme l'opération en une opération de mise à jour ayant les nouvelles clés et tente à nouveau l'appel.

Paramètre de procédure stockée qui indique si chaque colonne source a été supprimée

Une fois l'opération identifiée par le paramètre (`operation`), déclarez le second paramètre (`suppression_ind`) dans la procédure stockée comme paramètre IN indiquant si chaque colonne source a été supprimée.

Le paramètre IN `suppression_ind` est une matrice de caractères (`colsupind`) qui fait correspondre un caractère à chaque paramètre à l'exception des quatre paramètres obligatoires. Le caractère indique si la valeur de la colonne source a été supprimée :

- 1** La valeur de la colonne source en corrélation avec le paramètre a été supprimée.
- 0** La valeur de la colonne source en corrélation avec le paramètre n'a pas été supprimée

La matrice doit être utilisée au sein de la procédure stockée pour évaluer les paramètres de fonction. Le programme Q Apply effectue une vérification pour s'assurer qu'une seule valeur avant existe pour chaque paramètre qui mappe vers une colonne de clé.

Conseil : Voir les exemples de programmes pour obtenir des exemples de paramètre qui identifie des colonnes supprimées dans des procédures stockées écrites en langage C et SQL.

Paramètres de procédure stockée qui identifient la transaction pour la réplication Q

Une fois que le paramètre a identifié si les colonnes source ont été supprimées (`suppression_ind`), déclarez les deux paramètres suivants dans la procédure stockée comme paramètres IN qui recueillent des informations provenant de l'en-tête de message du programme Q Apply qui identifie la transaction. Puisque le programme Q Apply transmet des lignes et non pas des transactions à la procédure stockée, les paramètres transactionnels vous fournissent des informations pour vous aider à identifier la transaction à laquelle cette modification de ligne appartient de même que les horodatages et le numéro d'ordre du journal vous aident à établir un rapport et un diagnostic sur les erreurs.

Les deux paramètres suivants identifient la transaction :

IN `src_commit_lsn` CHAR(10)

Le numéro d'ordre du journal du moment où le serveur source a émis l'instruction de validation pour la transaction. Le type de données que vous indiquez pour ce paramètre doit être VARCHAR(16) si la base de données source est DB2 10.1 pour Linux, UNIX et Windows ou version ultérieure, et que la version du programme Q Capture est version 10.1 ou ultérieure avec la valeur du paramètre `compatibility` à 1001 ou supérieur.

IN `src_trans_time` TIMESTAMP

L'horodatage du moment où le serveur source a émis l'instruction de validation pour la transaction.

Conseil : Voir les exemples de programmes pour obtenir des exemples de deux paramètres transactionnels dans des procédures stockées écrites en langage C et SQL.

Paramètres de procédure stockée qui mappent vers des colonnes source pour la réplication Q

Après les paramètres de la procédure stockée qui identifient la transaction (`src_commit_lsn` et `src_trans_time`), déclarez les paramètres suivants de la procédure stockée comme paramètres IN qui recueillent les données source à partir de chaque colonne. Vous pouvez répertorier les paramètres supplémentaires qui mappent vers les colonnes dans n'importe quel ordre.

Après les paramètres opérationnels et transactionnels, incluez uniquement les paramètres de la procédure stockée qui mappent vers les valeurs TARGET_COLNAME dans la table IBMQREP_TRG_COLS. Aucun autre paramètre supplémentaire n'est autorisé dans la procédure stockée, même si ils sont déclarés en tant que paramètres OUT ou INOUT.

Les noms répertoriés dans la colonne TARGET_COLNAME doivent être les mêmes noms que les paramètres de la procédure stockée. Le tableau 20, à la page 189 illustre un exemple de la manière dont les noms de colonne source correspondent aux noms de paramètre dans la table IBMQREP_TRG_COLS. La table n'affiche que les trois colonnes pertinentes provenant de la table de contrôle.

Tableau 20. Les noms de colonne source et les noms de paramètres de la procédure stockée sont archivés dans la table IBMQREP_TRG_COLS

...	SOURCE_COLNAME	TARGET_COLNAME	IS_KEY	...
	Col1	Parm1	Y	
	Col2	Parm2	Y	
	Col3	Parm3	N	
	Col4	Parm4	N	
	Col5	Parm5	N	

Vous devez déclarer certains paramètres dans la procédure stockée pour faire partie de la clé. Les paramètres que vous déclarez en tant que colonnes de clé cible doivent mapper vers des colonnes de clé à la source ; les clés source et cible doivent correspondre. Le programme Q Apply utilise ces colonnes de clé pour correctement ordonner des transactions qui traitent les mêmes lignes.

Si le paramètre mappe vers une colonne de clé de la table source, vous devez alors effectuer les actions suivantes :

- Déclarez la colonne valeur avant correspondante comme paramètre.
- Assurez-vous que le paramètre de la valeur avant de la clé apparaît directement devant le paramètre de la valeur après de la clé.
- Faites commencer le nom du paramètre pour la valeur avant de la clé par un 'X' puis saisissez le nom du paramètre. Par exemple, pour la colonne de clé nommée Parm1, le paramètre de la valeur avant est XParm1.

Ne déclarez pas des paramètres pour des valeurs avant de colonnes de non clé.

Conseil : Si vous avez de nombreuses colonnes source que vous mappez vers des paramètres dans la procédure stockée, pensez à désigner les paramètres par les noms des colonnes source. Si les paramètres sont nommés de manière identique aux colonnes source auxquelles ils correspondent, alors le Centre de réplication est capable de mapper automatiquement les noms de colonne source vers les noms de paramètre cible.

Exemple de signature valide d'une procédure stockée avec des paramètres clés : dans cet exemple, Parm1 et Parm2 sont des paramètres qui mappent vers des colonnes clés.

```
CREATE PROCEDURE storedprocedure_name(
    INOUT operation integer,
    IN suppression_ind VARCHAR(size),
    IN src_commit_tsn CHAR(10),
    IN src_trans_time TIMESTAMP,
    XParm1,
    Parm1,
    XParm2,
    Parm2,
    Parm3,
    Parm4,
    Parm5
)
```

Exemple de procédure stockée non valide avec des paramètres clés : dans cet exemple non valide, Parm1 et Parm2 sont des paramètres qui mappent vers des colonnes clés. Cet exemple n'est pas valide car aucun paramètre n'accepte la valeur avant du paramètre de clé Parm2.

```

#-----
# Incorrect : à éviter
#-----

CREATE PROCEDURE storedprocedure_name(
    INOUT operation integer,
    IN suppression_ind VARCHAR(size) ,
    IN src_commit_lsn CHAR(10),
    IN src_trans_time TIMESTAMP,
    XParm1,
    Parm1,
    Parm2,
    XParm3,
    Parm3,
    Parm4,
    Parm5
)

```

Conseil : Voir les exemples de programmes pour obtenir des exemples de paramètres qui mappent vers les colonnes source dans les procédures stockées écrites en langage C et SQL.

Version 10.1 et ultérieure : Le paramètre de procédure stockée obligatoire `src_commit_lsn` doit indiquer un type de données `VARCHAR(16)` dans les conditions suivantes :

- La base de données source est DB2 10.1 pour Linux, UNIX et Windows, ou une version ultérieure
- Le programme Q Capture est la version 10.1 ou ultérieure et la valeur du paramètre **compatibility** est 1001 ou supérieur

Chapitre 11. Chargement des tables cibles pour la réplication Q

Lorsque vous créez un abonnement Q, vous pouvez choisir parmi plusieurs options pour charger des tables cible avec des données provenant de tables source.

Chargement automatique

Le programme Q Apply gère le chargement des tables cible. Vous pouvez sélectionner l'utilitaire que le programme Q Apply doit appeler ou laisser ce programme choisir le meilleur utilitaire disponible pour votre système d'exploitation et votre version.

Chargement manuel

Vous gérez le chargement des tables cible et informez les programmes de réplication lorsque le chargement est terminé. Cette option est également connue sous le nom de chargement externe.

Aucun chargement

Vous ne chargez aucune table cible, ou chargez ces tables hors du cadre des programmes de réplication.

Recommandations relatives au chargement de tables cible pour une réplication Q

La réplication Q peut être configurée de manière à ce que le chargement des tables cible s'effectue automatiquement, et est conçue pour se poursuivre sans perte de données pendant le processus de chargement.

Ces quelques recommandations garantiront le bon fonctionnement du processus :

Applications sur la cible

N'autorisez aucune application à mettre à jour les tables cible au cours de leur chargement. Les données de ces tables se révéleront incohérentes au cours du chargement ; le programme Q Apply supprime les contraintes d'intégrité référentielle jusqu'à ce que la table cible et tout autre table cible associée soient chargées.

Les applications peuvent réutiliser les tables cible en toute sécurité si le programme Q Apply effectue les modifications suivantes dans la table `IBMQREP_TARGETS` :

- Il indique la valeur A (actif) dans la colonne `STATE`.
- Il définit la valeur de la colonne `STATE_INFO` sur `ASN7606I`, ce qui signifie le rétablissement de toutes les contraintes d'intégrité référentielle sur la table cible, le cas échéant.

Vous pouvez utiliser la fenêtre Gestion des abonnements Q du Centre de réplication pour vérifier que ces modifications ont été effectuées. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture où la table source de l'abonnement Q est localisée puis sélectionnez **Gérer > Abonnements Q**.

Si le processus de chargement échoue ou si le programme Q Apply s'arrête en cours de chargement, toutes les données qui figuraient dans la table cible avant le début du chargement sont supprimées. Les modifications

effectuées dans la table source au cours du chargement ne sont pas perdues ; elles seront appliquées dans la table cible une fois le chargement correctement effectué.

Applications sur la source

Profitez de l'inactivité relative de la source pour charger les tables cible.

Option de chargement automatique pour la réplication Q

Vous pouvez laisser au programme Q Apply le soin de charger la table cible pour un abonnement Q lorsque cet abonnement est activé. Cette option, appelée *chargement automatique*, est sélectionnée par défaut pour la réplication Q.

Par défaut, lorsque vous optez pour le chargement automatique, le programme Q Apply choisit le meilleur utilitaire de chargement disponible pour votre système d'exploitation et votre version. Si vous le préférez, lorsque vous créez un abonnement Q, vous pouvez indiquer au programme Q Apply l'utilitaire de chargement à utiliser.

Lors du processus de chargement automatique, toutes les transactions source capturées et envoyées au programme Q Apply sont placées dans une file d'attente auxiliaire par ce programme, ce qui permet la poursuite de la réplication pendant le processus de chargement. Le programme Q Apply applique ces transactions une fois que la table cible est chargée.

Utilitaires employés pour l'option de chargement automatique de réplication Q

Si vous optez pour le chargement automatique, vous pouvez laisser le programme Q Apply sélectionner le meilleur utilitaire de chargement disponible ou vous pouvez spécifier un utilitaire.

La liste suivante répertorie les différents utilitaires de chargement disponibles.

LOAD à partir de CURSOR

Utilise une option de l'utilitaire DB2 LOAD pour déplacer des données de la table source dans la table cible sans créer de fichier exporté intermédiaire.

Vous pouvez modifier et utiliser les instructions SQL suivantes pour définir l'encapsuleur, le serveur, le mappage utilisateur et le pseudonyme requis pour l'utilitaire LOAD from CURSOR :

```
CREATE WRAPPER DRDA;  
CREATE SERVER MVS TYPE DB2/MVS VERSION 8 WRAPPER DRDA AUTHID AZUMA  
PASSWORD AZUMA OPTIONS (ADD DBNAME 'ONOGAWA', PASSWORD 'Y');  
CREATE USER MAPPING FOR USER  
SERVER MVS  
OPTIONS (REMOTE AUTHID 'AZUMA',  
REMOTE_PASSWORD 'AZUMA');  
CREATE NICKNAME T1NK FOR MVS.AZUMA.T1;
```

L'instruction suivante supprime le pseudonyme une fois le chargement terminé :

```
DROP NICKNAME T1NK;
```

Important : Si vous prévoyez d'utiliser l'utilitaire LOAD from CURSOR avec un pseudonyme pour effectuer un chargement à partir d'une source

DB2 version 9.7 ou ultérieure, vous devez vous assurer que l'option suivante est définie pour le serveur propriétaire du pseudonyme :

```
db2 alter server nom_serveur OPTIONS  
(ADD CONCURRENT_ACCESS_RESOLUTION 'W');
```

Utilitaires EXPORT et LOAD

Utilise une combinaison de l'utilitaire DB2 EXPORT et de l'utilitaire DB2 LOAD.

Utilitaires EXPORT et IMPORT

Utilise une combinaison de l'utilitaire DB2 EXPORT et de l'utilitaire DB2 IMPORT.

Important : Si vous prévoyez d'utiliser l'utilitaire DB2 EXPORT pour charger des tables cible à partir d'une source DB2 version 9.7 ou ultérieure et que l'ID utilisateur qui démarre le programme Q Apply ne dispose pas des droits BINDADD, vous devez effectuer la liaison suivante avant le démarrage de Q Apply :

```
db2 bind @db2ubind.lst CONCURRENTACCESSRESOLUTION WAIT FOR OUTCOME COLLECTION ASN
```

Si vous utilisez l'utilitaire EXPORT, le programme Q Apply demande un fichier de mots de passe pour se connecter au serveur Q Capture, à moins que les serveurs source et cible soient les mêmes. Pour créer un fichier des mots de passe, utilisez la commande **asnpwd**. Le fichier IXF ou fichier délimité par virgules est créé à l'emplacement défini par le paramètre **apply_path**.

z/OS Le programme Q Apply utilise LOAD from CURSOR sur DB2 version 7.1 pour z/OS ou ultérieure. InfoSphere Replication Server for z/OS 10.1 fonctionne uniquement avec DB2 for z/OS version 8 et plus.

Si vous définissez un utilitaire de chargement qui n'est pas disponible, le programme Q Apply interrompt l'abonnement Q.

Restrictions

Les restrictions suivantes s'appliquent à l'utilisation d'utilitaires pour des chargements automatiques :

Données d'objets LOB

Si vous répliquez à partir de tables comportant des données d'objet LOB et que les serveurs sont distants l'un de l'autre, EXPORT/LOAD n'est pas une option de chargement valide. Si vous utilisez une réplication bidirectionnelle ou une réplication entre homologues impliquant des tables avec des données d'objet LOB et des serveurs distants, l'utilitaire IMPORT n'est pas une option valide. Utilisez l'utilitaire LOAD from CURSOR dans ces deux situations.

Cibles de bases de données partitionnées avec des sources z/OS

Si vous effectuez une réplication à partir d'une source z/OS vers une base de données partitionnée sous Linux, UNIX ou Windows, les utilitaires EXPORT/LOAD et EXPORT/IMPORT ne sont pas pris en charge car l'utilitaire EXPORT à partir de z/OS ne prend en charge que le format de fichier IXF alors que l'utilitaire IMPORT/LOAD dans des bases de données partitionnées ne prend en charge que le format de fichier délimité par virgules. Pour un chargement automatique, vous devez utiliser l'utilitaire LOAD from CURSOR dans cette situation.

Considérations relatives au chargement automatique pour z/OS

Lorsqu'un programme Q Apply en cours d'exécution sur la plateforme z/OS procède à un chargement automatique des tables cible, il peut être opportun de configurer le paramètre NUMTCB et la taille des espaces table pour plusieurs chargements simultanés.

Configuration du paramètre NUMTCB

Le programme Q Apply utilise l'utilitaire LOAD à partir de CURSOR pour réaliser un chargement automatique des tables cible sur z/OS. Pour appeler cet utilitaire, le programme Q Apply appelle la procédure stockée DSNUTILS qui est fournie avec DB2 for z/OS.

DSNUTILS doit s'exécuter dans un environnement Work Load Manager (WLM). Vous devez configurer le paramètre NUMTCB, qui permet de démarrer WLM, en procédant de la manière suivante :

```
NUMTCB=1
```

Pour plus de détails sur DSNUTILS, consultez le *Guide de référence et des utilitaires* de DB2 for z/OS correspondant à votre version.

Remarques relatives aux espaces tables pour les chargements parallèles

Sous z/OS, si vous activez plusieurs abonnements Q simultanément et que le programme Q Apply est en train d'effectuer un chargement automatique des tables cible, le programme Q Apply chargera les tables cible en parallèle. Dans ce cas, vous devez vous assurer que chaque table cible réside dans un espace table distinct.

Au lieu de placer chaque table cible dans un espace table distinct, vous pouvez aussi démarrer chaque abonnement Q de manière séquentielle, de sorte que le chargement d'un abonnement Q ne démarre pas tant que celui de l'abonnement précédent n'est pas terminé.

Pour éviter les chargements parallèles :

1. Démarrez le premier abonnement Q.
2. Attendez que l'état de l'abonnement Q passe sur A (actif).

Vous pouvez vérifier l'état de l'abonnement Q à l'aide de la fenêtre Gérer les abonnements Q du Centre de réplication ou en consultant la colonne STATE de la table de contrôle IBMQREP_TARGETS.

3. Activez l'abonnement Q suivant.

Définition de pseudonymes pour l'option de chargement automatique de réplication Q

Certains abonnements Q qui utilisent l'utilitaire LOAD à partir de CURSOR pour charger des tables cible nécessitent des pseudonymes. Le pseudonyme est défini sur le serveur Q Apply pour renvoyer à la table source du serveur Q Capture.

Avant de commencer

- Le serveur Q Apply doit être un serveur fédéré.

- Si vous voulez que le centre de réplication ou le programme ASNCLP crée des pseudonymes, vous devez créer une définition de serveur, un encapsuleur et un mappage utilisateur.

A propos de cette tâche

Lorsque vous créez des abonnements Q, vous pouvez laisser le centre de réplication ou le programme ASNCLP créer des pseudonymes ou vous pouvez indiquer des pseudonymes existants.

Procédure

Vérification de la définition des paramètres d'accès simultané appropriés pour les pseudonymes utilisés pour le chargement

Si vous utilisez l'utilitaire LOAD from CURSOR pour charger des tables cible à partir d'une source DB2 version 9.7 ou ultérieure et que vous créez manuellement le pseudonyme utilisé pour le chargement, vous devez vérifier que le pseudonyme utilise l'option de serveur fédéré appropriée pour un accès simultané.

A propos de cette tâche

Depuis la version 9.7, une nouvelle option de serveur fédéré, `CONCURRENT_ACCESS_RESOLUTION=W`, est utilisée pour s'assurer que l'utilitaire LOAD from CURSOR attend que toutes les transactions en cours modifiant la table source soient terminées avant de commencer le chargement. Ce comportement est appelé "attente de résultat". Cette modification était nécessaire pour prendre en compte le comportement d'accès validé simultané par défaut dans DB2 pour Linux, UNIX et Windows, version 9.7 et ultérieures.

Prenez en compte les considérations suivantes :

- Si vous laissez le Centre de réplication ou le programme ASNCLP créer le pseudonyme utilisé pour l'utilitaire LOAD from CURSOR, l'option de serveur `CONCURRENT_ACCESS_RESOLUTION=W` est ajoutée au serveur pour ce pseudonyme. Dans certaines situations, il peut être nécessaire de créer manuellement le pseudonyme ou le pseudonyme utilisé pour le chargement peut être partagé avec d'autres applications. Dans ces situations, vous devez définir `CONCURRENT_ACCESS_RESOLUTION=W` manuellement pour le pseudonyme.
- Les procédures de définition de l'accès simultané sont différentes selon si le serveur Q Apply est à la version 9.7 ou une version ultérieure ou s'il est à une version antérieure à 9.7.
- **z/OS** Il n'existe actuellement aucune solution pour forcer le comportement d'attente de résultat lorsque Q Apply sous z/OS utilise l'utilitaire LOAD from CURSOR sur une base de données source DB2 V9.7 sous Linux, UNIX ou Windows pour effectuer le chargement. Dans ce cas, la meilleure solution consiste à interrompre toutes les applications mettant à jour la table source, entre le démarrage de l'abonnement Q et le début de la phase de chargement (l'état de l'abonnement Q passe alors à L ou A dans la table `IBMQREP_SUBS`).

Remarque : Vous ne pouvez définir cette option que pour un serveur enregistré de type DB2/UDB version 9.7 ou ultérieure.

Procédure

Pour vous assurer que les pseudonymes utilisés pour le chargement sont correctement configurés pour l'accès simultané, utilisez l'une des procédures suivantes, selon si le programme Q Apply est à la version 9.7 ou une version ultérieure, ou s'il est à une version antérieure à 9.7 :

Version du programme Q Apply	Procédure.
Version 9.7 ou ultérieure	Emettez la commande suivante sur le serveur Q Apply : <pre>db2 alter server <i>nom_serveur</i> OPTIONS (ADD CONCURRENT_ACCESS_RESOLUTION 'W');</pre>
Version antérieure à 9.7	<p>Remarque : Si vous ne parvenez pas à suivre cette procédure, interrompez toutes les applications mettant à jour la table source au début du chargement.</p> <ol style="list-style-type: none"> 1. A partir du serveur Q Apply, connectez-vous à la base de données source. 2. Liez les modules SQL utilisés pour les connexions d'interface CLI (Call Level Interface) avec une option de liaison générique dans un module spécifique à l'aide de la commande suivante : <pre>db2 bind @db2cli.lst generic "CONCURRENTACCESSRESOLUTION WAIT_FOR_OUTCOME" COLLECTION ASN</pre> 3. Ajoutez la paire nom-valeur suivante au fichier <code>db2cli.ini</code> dans la base de données fédérée, sous la section déclarant les options de la définition du serveur auquel le pseudonyme appartient : <pre>[<i>nom_source_données</i>] CURRENTPACKAGESET=ASN</pre> <p>Où <i>nom_source_données</i> correspond à la base de données source à laquelle les modules <code>db2cli.bnd</code> ont été liés.</p>

Recommandation : Si vous utilisez un serveur fédéré pour la réplication et pour d'autres activités, créez un nouveau serveur dédié pour la réplication, en définissant l'option `CONCURRENT_ACCESS_RESOLUTION=W` et en permettant à d'autres applications d'utiliser le nom de serveur existant.

Chargement manuel d'une table cible

Lorsque vous spécifiez l'option *chargement manuel* pour un abonnement Q, vous devez charger la table cible à l'aide de l'utilitaire de votre choix, puis avertir le programme Q Capture de la fin du chargement. Lorsque vous spécifiez l'option *chargement manuel* pour un abonnement Q, vous devez charger la table cible à l'aide de l'utilitaire de votre choix, puis avertir les composants de capture Classic de la fin du chargement.

Avant de commencer

- Veillez à ce que la colonne `HAS_LOADPHASE` dans la table `IBMQREP_SUBS` ait la valeur `E` pour indiquer qu'un chargement manuel sera effectué.
- Vérifiez qu'aucune application n'est en train de mettre à jour la table à charger. Lorsque les programmes de réplication synchroniseront les tables source et cible au terme du chargement, les données de ces tables seront incohérentes. Le programme Q Apply supprime les contraintes d'intégrité référentielle jusqu'à ce que la table cible et tout autre table cible associée soient chargées.

A propos de cette tâche

Au cours du chargement de la table cible, le programme Q Capture continue à envoyer des transactions depuis la table source. Le programme Q Apply les place dans une file d'attente auxiliaire temporaire, puis les applique une fois le chargement terminé. Il attend que tous les abonnements Q dépendants aient terminé leur phase de chargement avant de rétablir les contraintes d'intégrité référentielle sur la table cible.

Recommandation : Profitez de l'inactivité relative de la source pour charger les tables cible.

La figure 24 illustre les étapes du processus de chargement manuel.

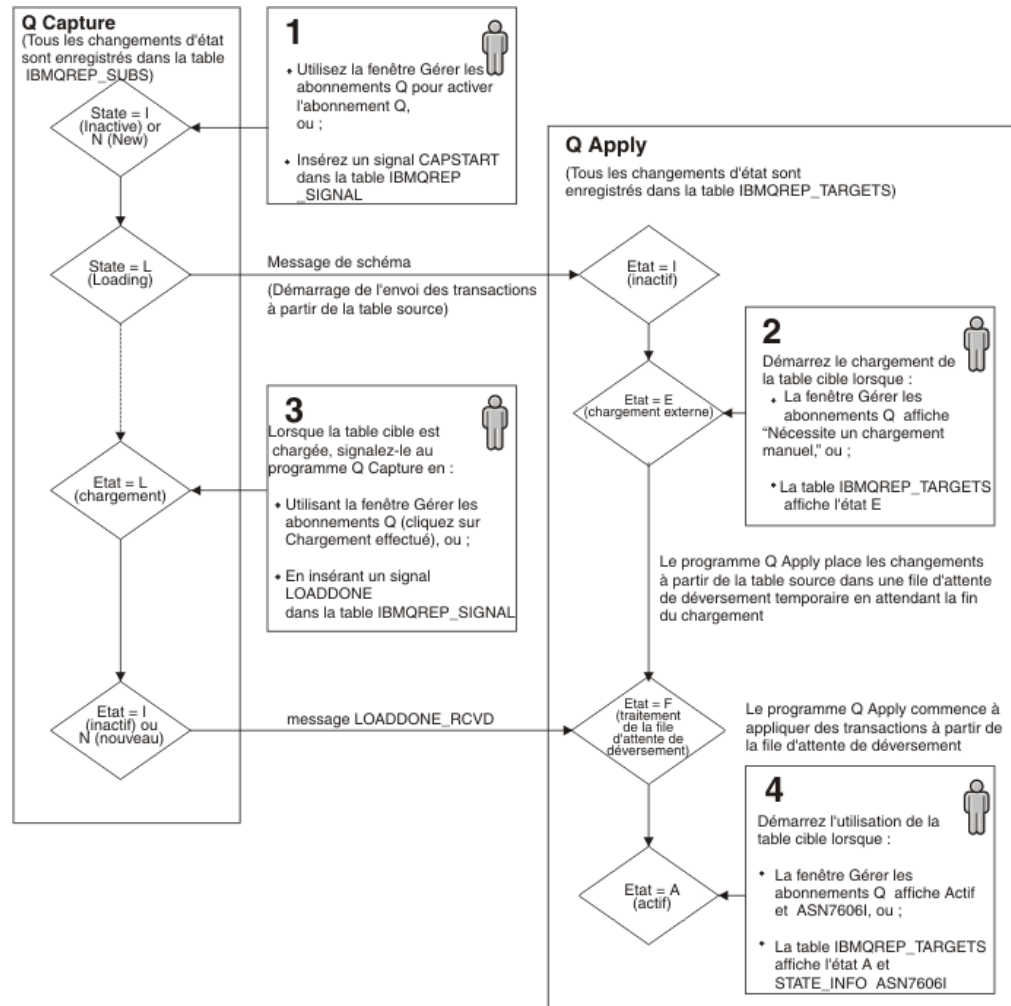


Figure 24. Etapes du processus de chargement manuel

Procédure

Pour charger manuellement une table cible :

- Démarrez l'abonnement Q à l'aide du centre de réplication ou de la commande **START QSUB**. Si vous avez choisi le démarrage automatique lors de la création de l'abonnement Q, son démarrage aura lieu à la réinitialisation du programme Q Capture.
- Vérifiez que le programme Q Apply attend la fin du chargement de la table cible. Utilisez l'une des méthodes suivantes :

Méthode	Description
Centre de réplication	Utilisez la fenêtre Gérer les abonnements Q. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture où la table source de l'abonnement Q est localisée puis sélectionnez Gérer > Abonnements Q . Recherchez l'abonnement dans la fenêtre, puis vérifiez que son état est sur "Nécessite un chargement manuel."
SQL	Lancez une instruction SELECT pour la table IBMQREP_TARGETS sur le serveur Q Apply, puis vérifiez que la valeur figurant dans la colonne STATE est E.

3. Chargez la table cible avec l'utilitaire de votre choix.
4. Avertissez le programme Q Capture lorsque la table est chargée. Utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande LOAD DONE. Par exemple, les commandes suivantes configurent l'environnement et génèrent le signal SQL permettant d'informer un programme Q Capture sur une base de données SAMPLE que le chargement est terminé pour l'abonnement Q EMPLOYEE0001 : ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE ASN1; LOAD DONE QSUB SUBNAME EMPLOYEE0001;
Centre de réplication	Dans la fenêtre Gestion des abonnements Q, sélectionnez l'abonnement correspondant à la table cible que vous avez chargée, puis cliquez sur Chargement terminé .
SQL	Pour insérer un signal LOADDONE dans la table IBMQREP_SIGNAL du serveur Q Capture, saisissez l'instruction suivante : insert into <i>schéma</i> .IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'LOADDONE', ' <i>sous-nom</i> ', 'P'); Où <i>schéma</i> identifie le programme Q Capture à avertir, et <i>sous-nom</i> est le nom de l'abonnement Q pour lequel vous effectuez un chargement manuel.

5. Pour vous assurer que les applications peuvent réutiliser les tables cible en toute sécurité, vérifiez si le programme Q Apply effectue les modifications suivantes dans la table IBMQREP_TARGETS :
 - Il indique la valeur A (actif) dans la colonne STATE.
 - Il définit la valeur de la colonne STATE_INFO sur ASN7606I, ce qui signifie le rétablissement de toutes les contraintes d'intégrité référentielle sur la table cible.

Vous pouvez utiliser la fenêtre Gestion des abonnements Q du Centre de réplication pour vérifier que ces modifications ont été effectuées. Il vous suffit de rechercher les valeurs Actif et ASN76061 dans la ligne correspondant à l'abonnement Q.

Option Aucun chargement pour la réplication Q

L'option Aucun chargement est appropriée lorsque les tables source et cible ont été synchronisées avant l'activation de tout abonnement Q et le début de toute réplication.

Lorsque vous spécifiez l'option aucun chargement pour un abonnement Q, le programme Q Apply commence à appliquer des transactions sur une table cible dès que l'abonnement devient actif.

Si vous choisissez cette option, assurez-vous que les valeurs de la clé primaire ou de l'index à entrées uniques de la table source figurent également dans la clé primaire ou l'index à entrées uniques de la table cible.

Vous pouvez spécifier l'option aucun chargement lors de l'ajout d'un grand nombre de nouvelles tables au cours d'une période d'inactivité relative dans les bases de données ou les sous-systèmes cible. Après avoir mis les tables source au repos, vous chargerez les tables cible et activerez les abonnements Q.

Cette option convient également lorsque vous sauvegardez une base de données source et la restaurez sur le serveur cible.

Options de chargement concernant différents types de réplication Q

Les options de chargement des tables cibles dépendent du type de réplication Q que vous configurez.

Les sections suivantes détaillent les options disponibles pour chaque type de réplication Q, les serveurs utilisés pour le chargement initial et les actions à effectuer pour initier le chargement :

- «Options de chargement pour une réplication unidirectionnelle»
- «Options de chargement pour une réplication bidirectionnelle ou entre homologues avec deux serveurs», à la page 200
- «Options de chargement pour une réplication entre homologues avec trois serveurs ou plus», à la page 202

Options de chargement pour une réplication unidirectionnelle

Cette section détaille les options de chargement concernant une réplication unidirectionnelle.

Options de chargement

Les options disponibles dépendent du type de cible :

Cibles DB2

Toutes les options de chargement sont disponibles.

Cibles CCD

Si la table CCD est incomplète, la seule option valide est Aucun chargement. Si la table est complète, toutes les options sont disponibles.

Cibles non-DB2

Pour les chargements automatiques, les utilitaires EXPORT et IMPORT sont pris en charge. Vous pouvez également indiquer que Q Apply utilise une sélection de connectivité ODBC pour extraire les données de la source ; pour les cibles Oracle, vous pouvez indiquer que Q Apply appelle l'utilitaire SQL*Loader.

Serveur utilisé pour le chargement initial

serveur Q Capture

Action à accomplir

Par défaut, le programme Q Apply initie le processus de chargement lorsque vous démarrez le programme Q Capture pour la table source de l'abonnement Q. Si vous avez créé cet abonnement de manière à ce qu'il ne démarre pas automatiquement au lancement du programme Q Capture, vous devez le démarrer pour initier le chargement.

Exemple

Supposons que vous souhaitiez répliquer des données dans un seul sens, de la table DEPARTMENT située sur le serveur A vers la table DEPARTMENT située sur le serveur B, à l'aide de la méthode la plus automatique. Vous souhaitez que le programme Q Apply gère le chargement et choisisse le meilleur utilitaire disponible.

1. Vous créez un abonnement Q pour la table DEPARTMENT en spécifiant l'option Chargement automatique qui choisit le meilleur utilitaire disponible.
2. Vous démarrez le programme Q Capture sur le serveur A et le programme Q Apply sur le serveur B.

Le programme Q Apply appelle un utilitaire de chargement qui copie les données de la table DEPARTMENT du serveur A vers la table DEPARTMENT du serveur B. Au terme de ce chargement, la réplication commence dans un sens, du serveur A vers le serveur B.

Options de chargement pour une réplication bidirectionnelle ou entre homologues avec deux serveurs

Cette section détaille les options de chargement concernant une réplication bidirectionnelle ou entre homologues avec deux serveurs.

Options de chargement

Toutes les options de chargement sont disponibles. Toutefois, si vous optez pour le chargement automatique, le programme Q Apply choisira par défaut l'une de ces deux combinaisons d'utilitaires, EXPORT et LOAD, ou EXPORT et IMPORT, en fonction de votre système d'exploitation et de votre version. Vous pouvez ignorer ce comportement et ordonner au programme Q Apply d'utiliser l'option LOAD FROM CURSOR, en ouvrant le bloc-notes Propriétés des abonnements Q pour chaque abonnement.

Serveur utilisé pour le chargement initial

Lorsque vous créez les deux abonnements Q pour la réplication bidirectionnelle ou entre homologues avec deux serveurs, vous choisissez le serveur qui constituera la source initiale du chargement. Ce serveur contient la table dont les données seront copiées vers l'autre serveur.

Pour les chargements suivants (par exemple, si vous arrêtez puis redémarrez les abonnements Q de la table logique), vous indiquerez le

serveur qui constituera la source du chargement au moment de choisir l'abonnement à démarrer. La table source de l'abonnement démarré constituera la source du chargement.

Action à accomplir

Le processus d'initiation du chargement dépend du type de chargement choisi, automatique ou manuel :

Chargement automatique

Si vous avez créé les abonnements Q de sorte qu'ils soient démarrés automatiquement au lancement du programme Q Capture, il vous suffit de démarrer les programmes Q Apply et Q Capture sur les deux serveurs pour initier le chargement.

Si vous n'avez pas opté pour le démarrage automatique de l'abonnement, vous devez accomplir les actions suivantes :

- Démarrez les programmes Q capture et Q apply sur les deux serveurs.
- Démarrez l'abonnement Q correspondant à la table source désignée comme source du chargement.

Chargement manuel

1. Démarrez les programmes Q capture et Q apply sur les deux serveurs.
2. Démarrez l'abonnement Q correspondant à la table source que vous souhaitez désigner comme source du chargement. L'abonnement prendra l'état de chargement en attente.
3. Chargez la table cible de l'abonnement Q, par tout moyen choisi.
4. Lorsque le chargement est terminé, avertissez le Centre de réplication ou insérez un signal LOADDONE dans la table IBMQREP_SIGNAL située sur le serveur source de l'abonnement.

Exemple

Supposons que vous souhaitiez répliquer la table EMPLOYEE de manière bidirectionnelle sur les serveurs A et B, en recourant à la méthode la plus automatique. Vous souhaitez que le serveur A soit la source du chargement :

1. Vous créez deux abonnements Q : EMP_A2B et EMP_B2A. En créant l'abonnement EMP_A2B, vous désigner le serveur A comme source initiale du chargement et optez pour le chargement automatique, en vertu duquel le programme Q Apply choisit le meilleur utilitaire de chargement disponible.
2. Vous initiez le chargement en démarrant les programmes Q Capture et Q Apply sur les deux serveurs.

Le programme Q Apply du serveur B lance le chargement pour l'abonnement EMP_A2B. Pour ce faire, il appelle un utilitaire de chargement qui va copier les données de la table EMPLOYEE du serveur A vers la table EMPLOYEE du serveur B. Au terme de ce chargement, la réplication commence dans les deux sens, entre le serveur A et le serveur B.

Options de chargement pour une réplication entre homologues avec trois serveurs ou plus

Cette section détaille les options de chargement concernant une réplication entre homologues avec trois serveurs ou plus.

Options de chargement

Toutes les options de chargement sont disponibles. Toutefois, si vous optez pour le chargement automatique, le programme Q Apply choisit par défaut l'une de ces deux combinaisons d'utilitaires, EXPORT et LOAD, ou EXPORT et IMPORT, en fonction de votre système d'exploitation et de votre version. Vous pouvez ignorer ce comportement et ordonner au programme Q Apply d'utiliser l'option LOAD FROM CURSOR, en ouvrant le bloc-notes Propriétés des abonnements Q pour chaque abonnement.

Serveur utilisé pour le chargement initial

Dans un groupe d'homologues comptant trois serveurs ou plus, le lancement de la réplication s'effectue par étapes. Vous démarrez d'abord la réplication entre deux serveurs, puis intégrez des serveurs supplémentaires en lançant une réplication entre un serveur actif et un nouveau serveur. Suivez ces instructions :

- Lorsque vous démarrez la réplication entre les deux premiers serveurs, désignez l'un d'eux comme source du chargement. Démarrez l'abonnement Q qui spécifie ce serveur comme étant sa source. Le programme Q Apply du second serveur entame le processus de chargement de la table sur son serveur.
- Pour ajouter un nouveau serveur, désignez l'un des serveurs actifs comme source du chargement. Démarrez l'abonnement Q qui spécifie ce serveur comme étant sa source et le nouveau serveur comme étant sa cible. Le programme Q Apply du nouveau serveur entame le processus de chargement de la table sur son serveur.

Action à accomplir

Dans une configuration entre homologues comportant trois serveurs ou plus, vous ne pouvez pas créer d'abonnement avec démarrage automatique. Vous devez démarrer les abonnements manuellement et par étapes. Procédez comme suit :

1. Démarrez les programmes Q Capture et Q Apply sur les deux premiers serveurs du groupe.
2. Démarrez l'un des deux abonnements Q entre ces serveurs. La table source de l'abonnement démarré constituera la source du chargement, et la table cible sera chargée.
3. Démarrez les programmes Q Capture et Q Apply sur un nouveau serveur.
4. Démarrez un abonnement Q qui spécifie l'un des serveurs actifs comme étant sa source et le nouveau serveur comme étant sa cible. La table source de l'abonnement démarré constituera la source du chargement, et la table du nouveau serveur sera chargée.
5. Répétez les étapes 3 et 4 jusqu'à ce que les chargements soient effectués sur tous les serveurs du groupe.

Chargement manuel : Si vous optez pour le chargement manuel, vous devez charger la table cible après avoir démarré chaque abonnement Q, puis avertir les programmes de réplication de la fin du chargement.

Exemple

Supposons que vous souhaitiez initier le processus de chargement pour un groupe d'abonnements Q entre homologues, qui inclut les serveurs A, B et C, et concerne une seule table logique, la table DEPARTMENT. Vous souhaitez que le programme Q Apply gère le chargement et choisisse le meilleur utilitaire disponible. Vous utiliserez le serveur A en tant que source du chargement, tant sur le serveur B que sur le serveur C.

1. Vous créez six abonnements Q : DEP_A2B, DEP_B2A, DEP_A2C, DEP_C2A, DEP_B2C, et DEP_C2B, tous devant être chargés automatiquement avec le meilleur utilitaire disponible.
2. Vous démarrez les programmes Q Capture et Q Apply sur les serveurs A et B.
3. Vous démarrez l'abonnement DEP_A2B.

Le programme Q Apply du serveur B appelle un utilitaire qui charge la table DEPARTMENT sur ce serveur en copiant les données de la table DEPARTMENT du serveur A. Au terme de ce chargement, la réplication commence dans les deux sens, entre le serveur A et le serveur B.

4. Pour initier le chargement sur le serveur C, vous devez d'abord démarrer les programmes Q Capture et Q Apply sur ce serveur.
5. Ensuite, vous démarrez l'abonnement DEP_A2C.

Le programme Q Apply du serveur C appelle un utilitaire qui charge la table DEPARTMENT sur ce serveur en copiant les données de la table DEPARTMENT du serveur A. Au terme de ce chargement, la réplication commence dans tous les sens, entre les trois serveurs.

Réplication d'opérations de chargement de la table source

Vous pouvez spécifier que les opérations de chargement de la table source effectuées à l'aide de l'utilitaire LOAD DB2 sont répliquées sur la table cible.

A propos de cette tâche

Par défaut, lorsque le programme Q Capture lit un enregistrement de journal qui indique que la table source a été chargée avec succès, il émet un message d'avertissement. Pour modifier ce comportement par défaut, au moment de la création d'un abonnement Q pour une table source, indiquez que Q Capture réplique certains types d'opérations de chargement sur la table source.

Lorsque cette fonction est activée, Q Capture arrête et démarre l'abonnement Q de la table source, ce qui entraîne le chargement de la table cible si cette dernière est définie pour l'abonnement Q, cette opération s'effectue en fonction des options de chargement qui ont été définies pour l'abonnement Q.

Lorsque Q Capture détecte les opérations DB2 suivantes, ses actions ultérieures varient selon que vous avez activé ou non la réplication des chargements de table source :

z/OS

- LOAD SHRLEVEL NONE RESUME YES
- LOAD SHRLEVEL NONE REPLACE
- REORG TABLESPACE DISCARD

Linux UNIX Windows

- LOAD REPLACE

- LOAD INSERT

Les options suivantes utilisées dans le cadre de la réplication de chargements de table source sont stockées dans la colonne CAPTURE_LOAD de la table IBMQREP_SUBS :

R (redémarrer)

L'abonnement Q est redémarré et la table cible est chargée avec les données issues de la table source. Le type de chargement est déterminé par la valeur LOAD_TYPE dans la table de contrôle IBMQREP_TARGETS.

W (avertissement)

Seul un message d'avertissement est émis. L'abonnement Q n'est pas redémarré.

Restrictions

- **z/OS** La base de données source DB2 for z/OS version 9 doit être au niveau de correctif APAR PK78558 DB2 ou plus récent, et le programme Q Capture utilisé doit être au niveau de version 9.7 ou ultérieur. De plus, cette fonction n'est pas prise en charge si la table source ou cible partage un espace table avec d'autres tables. CAPTURE_LOAD est une option au niveau de l'espace table.
- **Linux UNIX Windows** La base de données source DB2 for Linux, UNIX et Windows et le programme Q Capture doivent être au niveau de version 9.7 ou ultérieur.
- La réplication des chargements de la table source n'est pas prise en charge pour la réplication entre homologues avec trois serveurs ou plus.
- Lorsque vous chargez une table impliquée dans une réplication bidirectionnelle, vous devez laisser se terminer la réplication de cette opération de chargement et l'état de l'abonnement Q passer à actif (A) avant de commencer une autre opération de chargement de la table correspondante sur l'autre serveur. Sinon, les deux opérations de chargement entreraient en conflit et pourraient générer un comportement imprévu.
- Le chargement d'une table source alors que des abonnements Q pour la table sont toujours à l'état de chargement (par exemple, deux opérations de chargement consécutives) provoque l'arrêt des abonnements Q pour la table. Vous devez toujours attendre que tous les abonnements Q pour la table source soient à l'état actif (A) avant d'effectuer une nouvelle opération de chargement.

Procédure

Pour répliquer des opérations de chargement de la table source, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Dans la commande CREATE QSUB, définissez le mot clé CAPTURE_LOAD avec l'option R, comme dans l'exemple suivant : CREATE QSUB USING REPLQMAP SAMPLE_ASN_TO_TARGET_ASN (SUBNAME EMPLOYEE0001 EMPLOYEE OPTIONS HAS LOAD PHASE I CAPTURE_LOAD R TARGET NAME TGTEMPLOYEE LOAD TYPE 2);
Centre de réplication	Cochez la case Recharger la table cible si la table source est rechargée de la page Chargement des tables cible de l'assistant Créer des abonnements Q. Cette commande est également disponible dans l'onglet Chargement du bloc-notes Propriétés des abonnements Q.

Comment les contraintes sur la table source affectent la réplication des opérations de chargement

La réplication des opérations de chargement sur la table source exige une attention particulière lorsque les contraintes sont définies dans la table source.

Une fois la table source chargée, l'accès en lecture requis par l'utilitaire LOAD DB2 pour charger les données dans la table cible peut être limité :

z/OS

Sous z/OS, l'accès peut être limité si des contraintes sont définies sur la table source et que l'utilitaire LOAD est appelé avec l'option REPLACE ou RESUME ENFORCE NO. Cette situation indique à l'utilitaire LOAD de mettre la table à l'état CHECK PENDING et Q Apply arrête l'abonnement Q. Pour corriger ce problème, vous devez exécuter CHECK DATA sur la table source pour rentrer l'espace table accessible puis redémarrer l'abonnement Q.

Linux UNIX Windows

Sous Linux, UNIX et Windows, l'accès peut être limité si des contraintes sont définies sur la table source et que l'utilitaire LOAD est appelé avec l'option REPLACE ou INSERT NO ACCESS. Cette situation indique à l'utilitaire LOAD de mettre la table à l'état SET INTEGRITY CHECK PENDING et Q Apply arrête l'abonnement Q. Pour corriger ce problème, vous devez exécuter SET INTEGRITY sur la table source pour rentrer l'espace table accessible puis redémarrer l'abonnement Q.

Avertissement : L'option INSERT ALLOW READ ACCESS de l'utilitaire LOAD n'est pas recommandée si vous avez défini l'option sur les chargements de table source de réplication. Si l'utilitaire est appelé avec cette option et que des contraintes sont définies sur la table source, la table est mise à l'état SET INTEGRITY PENDING et READ ACCESS. Dans ces deux états, les données qui viennent d'être chargées ne sont pas accessibles. SI vous prévoyez d'utiliser cette option, utilisez l'une des méthodes suivantes pour éviter la perte de la synchronisation des tables source et cible :

- Avant le début du chargement, arrêtez l'abonnement Q, chargez la table source puis démarrez l'abonnement Q. Cette procédure déclenche un nouveau chargement complet de la table cible (actualisation intégrale).
- Définissez l'option CAPTURE_LOAD de l'abonnement Q sur W et chargez la table source pendant que l'abonnement Q est actif. Aucune des données venant d'être chargées n'est répliquée et vous pouvez alors effectuer l'une des tâches suivantes :
 - Utilisez les commandes asntdiff et asntrep pour rechercher et résoudre les différences entre les tables source et cible.
 - Arrêtez et démarrez l'abonnement Q, ce qui déclenche un chargement automatique de la cible.

Chargement de tables cible dans un scénario de consolidation de données

Lorsque plusieurs tables source sont répliquées dans une même table cible dans une réplication Q, vous utilisez une procédure spéciale qui s'assure que toutes les données des différents chargements sont conservées dans la table cible.

A propos de cette tâche

Cette procédure vous permet de charger de façon incrémentielle la cible à partir de plusieurs sources pour s'assurer qu'aucun téléchargement n'écrase les données des chargements précédents.

Lorsque vous créez des abonnements Q pour des tables source, vous définissez un type de chargement qui utilise l'option REPLACE pour un abonnement Q et un type de chargement qui utilise l'option RESUME (z/OS) ou INSERT (Linux, UNIX et Windows) pour tous les autres abonnements Q. Puis, vous démarrez d'abord l'abonnement Q avec le type de chargement REPLACE, puis les autres abonnements Q, lorsque le premier chargement est terminé.

Le premier chargement remplace toutes les données de la table cible et les chargements suivants ajoutent des données provenant des autres tables source sans affecter les données du premier chargement.

Le tableau 21 présente les types de chargement que vous devez utiliser en combinaison pour cette procédure. Les informations sur le type de chargement sont stockées dans la table IBMQREP_TARGETS.

Tableau 21. Charger des combinaisons de types pour la consolidation des données

Charger le type pour le premier chargement (avec l'option REPLACE)	Charger le type pour les chargements ultérieurs (avec l'option RESUME ou INSERT)
1 (LOAD from CURSOR)	101

Tableau 21. Charger des combinaisons de types pour la consolidation des données (suite)

Charger le type pour le premier chargement (avec l'option REPLACE)	Charger le type pour les chargements ultérieurs (avec l'option RESUME ou INSERT)
2 (EXPORT/IMPORT)	102
3 (EXPORT/LOAD)	103

La figure 25 affiche le fonctionnement de la procédure pour l'une de ces combinaisons, LOAD from CURSOR avec l'option REPLACE pour le premier chargement et LOAD from CURSOR avec l'option RESUME pour les chargements suivants.

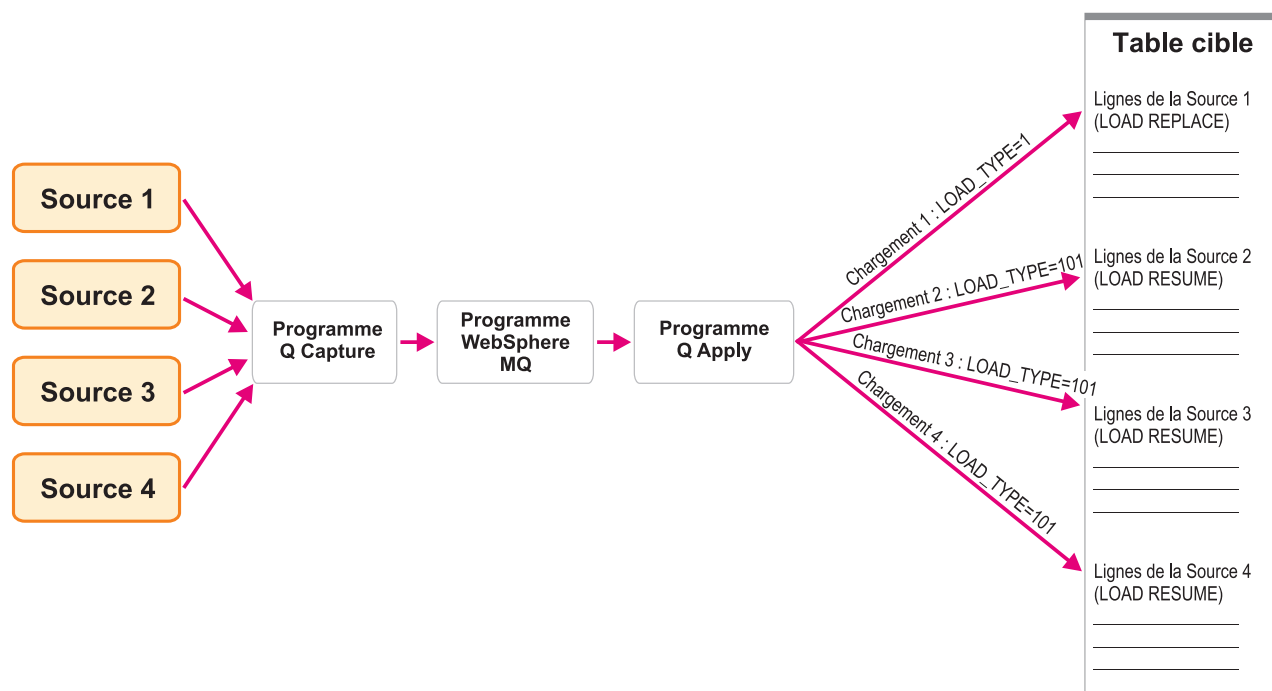


Figure 25. Exemple de procédure de chargement pour un scénario de consolidation des données qui utilise LOAD from CURSOR

Procédure

1. Lorsque vous créez des abonnements Q pour la table cible consolidée, définissez le LOAD_TYPE sur 1, 2, ou 3 pour un abonnement Q dans le groupe et le LOAD_TYPE sur 101, 102, ou 103 pour tous les abonnements Q restants du groupe, suivant les combinaisons présentées dans le tableau 21, à la page 206.

Remarque : Indiquez que les abonnements Q avec le LOAD_TYPE 101, 102, ou 103 sont créés à l'état inactif (I). Vous pouvez le faire dans le programme de ligne de commande ASNCLP à l'aide des mots clés START AUTOMATICALLY NO. Dans le Centre de réplcation, sur la page Chargement des tables cible de l'assistant Créer des abonnements Q, décochez la case **Démarrer automatiquement tous les abonnements Q**. L'abonnement Q avec le LOAD_TYPE 1, 2, ou 3 démarre automatiquement.

Ne définissez pas non plus l'action de conflit et l'action d'erreur pour les abonnements Q sur D (arrêt de l'abonnement Q).

2. Une fois le chargement terminé pour le premier abonnement Q, démarrez les autres abonnements Q.

Le programme Q Apply définit l'état de l'abonnement Q sur A (actif) lorsqu'un chargement est terminé. Vous pouvez utiliser le tableau de bord de réplication Q ou la fenêtre Gestion des abonnements Q du Centre de réplication pour vérifier l'état de l'abonnement Q.

Si la table cible est vide ou si vous supprimez manuellement toutes les lignes, vous pouvez définir le type LOAD_TYPE sur 101, 102 ou 103 pour tous les abonnements Q du groupe et les démarrer un par un.

Si vous ne devez recharger que les données d'une seule table source (si un abonnement Q a rencontré une erreur par exemple), supprimez manuellement toutes les lignes de la table cible en utilisant un intervalle adapté, définissez le type LOAD_TYPE sur 101, 102 ou 103 pour l'abonnement Q ayant rencontré une erreur et démarrez l'abonnement Q.

Si vous devez effectuer une régénération intégrale de l'intégralité de la table cible, suivez la procédure ci-dessus.

Réplication Q et publication d'événements pour plusieurs partitions de base de données

La réplication Q et la publication d'événements prennent en charge la capture des données à partir des tables source DB2 qui s'étendent sur plusieurs partitions de base de données.

Lorsque vous créez des tables de contrôle Q Capture dans une base de données à plusieurs partitions, tous les espaces table utilisés par ces tables de contrôle doivent se trouver sur la partition de catalogue. Si vous utilisez un espace table existant, il ne doit pas être partitionné et se trouver sur le noeud catalogue.

Le programme Q Capture conserve une liste des partitions de base de données dans le message de redémarrage. Chaque fois que le programme Q Capture fait un démarrage à chaud, il lit la liste des partitions de base de données à partir du message de redémarrage. Le programme Q Capture compare le nombre de partitions de base de données connues de DB2 au nombre de partitions de base de données répertoriées dans le message de redémarrage. Si les nombres ne correspondent pas, le programme Q Capture s'arrête.

Si vous avez ajouté une ou plusieurs partitions de base de données depuis la dernière exécution du programme Q Capture, vous devez indiquer à celui-ci ces nouvelles partitions. Vous pouvez le faire en démarrant Q Capture avec le paramètre **add_partition**.

Par exemple, la commande suivante spécifie qu'un programme Q Capture sur le serveur SAMPLE démarre la lecture du fichier journal pour les partitions récemment ajoutées :

```
asncap capture_server=sample capture_schema=asn1 add_partition=y
```

Réplication des tables partitionnées DB2 : Version 9.7 groupe de correctifs 1 et antérieurs (Linux, UNIX, Windows)

La réplication Q prend en charge les tables DB2 partitionnées par intervalle (en utilisant la clause PARTITION BY de l'instruction CREATE TABLE). Ces tables sont parfois appelées tables partitionnées par intervalle.

Des exigences de version et de groupe de correctifs existent pour le programme Q Capture si une table source est partitionnée par intervalle. Cette rubrique décrit ces exigences ainsi que la prise en charge de la réplication des tables partitionnées lorsque la version de vos programmes de réplication est 9.7 avec groupe de correctifs 1 ou version précédente. Si seules les tables cibles (aucune table source) sont partitionnées par intervalle, la réplication Q ne comporte aucune exigence de version ou de groupe de correctifs spécifique à ces tables.

Pour capturer les données modifiées pour les tables partitionnées par intervalle, votre programme Q Capture doit être à la version 9.7 ou ultérieure. Ce programme permet de capturer des modifications à partir des tables partitionnées par intervalle sur DB2 version 9.1, 9.5 ou 9.7. Cependant, il existe des restrictions pour les tables partitionnées par intervalle de version inférieure à 9.7. Ces restrictions sont également décrites dans cette rubrique.

Les tables partitionnées suivent un schéma d'organisation des données dans lequel les données de la table sont partagées entre plusieurs objets de stockage, appelés des partitions ou des intervalles de données, en fonction des valeurs contenues dans une ou plusieurs colonnes clés de partitionnement de la table.

La réplication et la publication traitent toutes les partitions de données d'une table source comme une seule table. Par exemple, lorsque vous créez un abonnement Q ou une publication qui spécifie une table partitionnée, vous indiquez toute la table et non une ou plusieurs partitions de données de celle-ci. Toutes les opérations sur la ligne effectuées pour la table, quelle que soit la partition de données où elles sont effectuées, sont répliquées ou publiées.

Vous pouvez apporter des modifications à une table partitionnée, notamment ajouter une partition de données, joindre ou déconnecter une partition. Ces opérations ALTER sur la table source ne sont pas répliquées sur la cible. Vous devez modifier la table cible indépendamment de la table source si vous voulez gérer un schéma de partitionnement identique.

La réplication et la publication traitent ces opérations ALTER différemment :

Remarque relative à la version 9.5 et antérieures : Le programme Q Capture ne reconnaît pas l'ajout, la connexion ou déconnexion d'une partition de données tant que le programme n'a pas été réinitialisé ou arrêté et redémarré.

ADD Ajoute une nouvelle partition de données, vide, à la table source. Si vous voulez ajouter cette nouvelle partition de données sur la cible, vous devez l'ajouter manuellement. Le comportement du programme Q Capture et la procédure que vous devez suivre dépendent de l'édition de votre DB2 :

Version 9.7 ou ultérieure

Ajout de la partition de données à la cible avant de l'ajouter à la table source. Q Capture commence automatiquement à répliquer les modifications sur la partition de données.

Version 9.5 ou 9.1

Ajoutez la partition de données sur la source et la cible avant de redémarrer Q Capture. Ne modifiez pas les données dans la partition de données source jusqu'à ce que Q Capture soit redémarré.

ATTACH

Crée une nouvelle partition de données sur la source à l'aide d'une table existante. L'opération ATTACH n'est pas répliquée et les données de la nouvelle partition de données ne sont pas répliquées sur la cible. Si vous voulez ajouter cette nouvelle partition de données sur la cible, vous devez l'ajouter manuellement. Si vous avez besoin des données jointes sur la cible, vous devez les charger manuellement dans la cible avant de joindre la partition de données à la cible.

Remarque : Si le programme Q Capture est arrêté lorsqu'une partition de données est jointe, les lignes qui sont insérées, mises à jour ou supprimées sur la table avant qu'elle ne soit jointe sont répliquées. Si Q Capture est en cours de fonctionnement lorsque la partition de données est jointe, ces lignes ne sont pas répliquées.

Pour garantir un comportement cohérent, assurez-vous que la clause DATA CAPTURE CHANGES pour la table est définie sur OFF si vous avez besoin d'apporter des modifications à la table. Par exemple, les instructions suivantes créent une table, insèrent des valeurs dans la table puis joignent la table comme partition de données à une table partitionnée existante :

```
db2 create table temp1 like t1;
-- NOTE: data capture changes is off by default
db2 insert into temp1 values (44,44);
-- NOTE: Turn on data capture changes after insert/update/deletes
-- and before attach partition
db2 alter table temp1 data capture changes;
db2 alter table t1 attach partition part4 starting from 41
ending at 50 from temp1;
db2 set integrity for t1 allow write access immediate checked;
```

DETACH

Transforme une partition de données existante en une table séparée. L'opération DETACH n'est pas répliquée. Les données qui sont supprimées de la table source via l'opération DETACH, ne sont pas supprimées de la table cible. Si vous devez modifier la partition de données cible en table séparée, vous devez le faire manuellement.

Remarque : DB2 consigne les mises à jour qui provoquent le déplacement des lignes sur les partitions de données en tant que paires supprimer/insérer. L'option d'abonnement Q ou de publication permettant d'éliminer les opérations de suppression de la table source (SUPPRESS_DELETES=Y) peuvent se traduire par le fait qu'une opération UPDATE unique à la source engendre deux lignes à la cible. Il est recommandé d'éviter d'utiliser l'option permettant d'éliminer les lignes supprimées avec des tables partitionnées.

Réplication des tables partitionnées DB2 : Version 9.7 groupe de correctifs 2 et ultérieurs (Linux, UNIX, Windows)

La réplication Q prend en charge les tables DB2 partitionnées par intervalle (en utilisant la clause PARTITION BY de l'instruction CREATE TABLE). Ces tables sont parfois appelées tables partitionnées par intervalle.

Des exigences de version et de groupe de correctifs existent pour le programme Q Capture si une table source est partitionnée par intervalle. Cette rubrique décrit ces exigences ainsi que la prise en charge de la réplication des tables partitionnées lorsque la version de vos programmes de réplication est 9.7 avec groupe de correctifs 2 ou ultérieur. Si seules les tables cibles (aucune table source) sont partitionnées par intervalle, la réplication Q ne comporte aucune exigence de version ou de groupe de correctifs spécifique à ces tables.

Pour capturer les données modifiées pour les tables partitionnées par intervalle, votre programme Q Capture doit être à la version 9.7 ou ultérieure. Ce programme permet de capturer des modifications à partir des tables partitionnées par intervalle sur DB2 version 9.1, 9.5 ou 9.7. Cependant, il existe des restrictions pour les tables partitionnées par intervalle de version inférieure à 9.7. Ces restrictions sont également décrites dans cette rubrique.

Important : Si vos programmes de réplication sont à la Version 9.7 groupe de correctifs 2 ou ultérieur, et que vous prévoyez de répliquer des tables partitionnées par intervalle, vous devez exécuter le script de migration de la Version 9.7 groupe de correctifs 2, `asnqcap1uwv97fp2.sql`. Le script ajoute une nouvelle table de contrôle, `IBMQREP_PART_HIST`, pour aider les programmes de réplication à traiter les changements de partition de données, tels que ajouter, joindre ou déconnecter. Le script est situé dans le répertoire `sql1lib/samples/rep1/mig97/q/`. Le programme Q Capture n'utilise pas la table `IBMQREP_PART_HIST` pour les tables de source partitionnées sur DB2 Version 9.5 ou Version 9.1.

Les tables partitionnées suivent un schéma d'organisation des données dans lequel les données de la table sont partagées entre plusieurs objets de stockage, appelés des partitions ou des intervalles de données, en fonction des valeurs contenues dans une ou plusieurs colonnes clés de partitionnement de la table.

La réplication et la publication traitent toutes les partitions de données d'une table source comme une seule table. Par exemple, lorsque vous créez un abonnement Q ou une publication qui spécifie une table partitionnée, vous indiquez toute la table et non une ou plusieurs partitions de données de celle-ci. Toutes les opérations sur la ligne effectuées pour la table, quelle que soit la partition de données où elles sont effectuées, sont répliquées ou publiées.

Vous pouvez apporter des modifications à une table partitionnée, notamment ajouter une partition de données, joindre ou déconnecter une partition. Ces opérations ALTER sur la table source ne sont pas répliquées sur la cible. Vous devez modifier la table cible indépendamment de la table source si vous voulez gérer un schéma de partitionnement identique.

La réplication et la publication traitent ces opérations ALTER différemment :

Remarque relative à la version 9.5 et antérieures : Le programme Q Capture ne reconnaît pas l'ajout, la connexion ou déconnexion d'une partition de données tant que le programme n'a pas été réinitialisé ou arrêté et redémarré.

ADD Ajoute une nouvelle partition de données, vide, à la table source. Si vous voulez ajouter cette nouvelle partition de données sur la cible, vous devez l'ajouter manuellement. Le comportement du programme Q Capture et la procédure que vous devez suivre dépendent de l'édition de votre DB2 :

Version 9.7 ou ultérieure

Ajout de la partition de données à la cible avant de l'ajouter à la

table source. Q Capture commence automatiquement à répliquer les modifications sur la partition de données.

Version 9.5 ou 9.1

Ajoutez la partition de données sur la source et la cible avant de redémarrer Q Capture. Ne modifiez pas les données dans la partition de données source jusqu'à ce que Q Capture soit redémarré.

ATTACH

Crée une nouvelle partition de données sur la source à l'aide d'une table existante. L'opération ATTACH n'est pas répliquée et les données de la nouvelle partition de données ne sont pas répliquées sur la cible. Si vous voulez ajouter cette nouvelle partition de données sur la cible, vous devez l'ajouter manuellement. Si vous avez besoin des données jointes sur la cible, vous devez les charger manuellement dans la cible avant de joindre la partition de données à la cible.

DETACH

Transforme une partition de données existante en une table séparée. L'opération DETACH n'est pas répliquée. Les données qui sont supprimées de la table source via l'opération DETACH, ne sont pas supprimées de la table cible. Si vous devez modifier la partition de données cible en table séparée, vous devez le faire manuellement.

Remarque : DB2 consigne les mises à jour qui provoquent le déplacement des lignes sur les partitions de données en tant que paires supprimer/insérer. L'option d'abonnement Q ou de publication permettant d'éliminer les opérations de suppression de la table source (SUPPRESS_DELETES=Y) peuvent se traduire par le fait qu'une opération UPDATE unique à la source engendre deux lignes à la cible. Il est recommandé d'éviter d'utiliser l'option permettant d'éliminer les lignes supprimées avec des tables partitionnées.

Tables CCD dans la réplication Q

Les tables CCD (Consistent-Change-Data) fournissent les données de transactions validées qui peuvent être lues et utilisées par d'autres applications, par exemple InfoSphere DataStage ou le programme Apply pour la réplication SQL.

En utilisant une table CCD comme type cible, vous pourrez conserver un historique des modifications apportées à la source. Par exemple, vous pouvez effectuer un suivi des comparaisons avant et après des données, le moment où des modifications sont apportées ainsi que l'ID utilisateur ayant mis à jour la table source.

Dans le cadre de la réplication Q unidirectionnelle, vous pouvez indiquer des tables CCD comme sources et cibles. Vous pouvez également indiquer une table CCD remplie par les programmes de réplication Q en tant que source pour la réplication SQL.

Les sections suivantes contiennent des informations détaillées sur les tables CCD dans la réplication Q :

- «Attributs Condensée et complète», à la page 213
- «Options de chargement des tables CCD», à la page 213
- «Options concernant les erreurs et les conflits», à la page 214
- «Colonnes par défaut des tables CCD», à la page 214

- «Colonnes de contrôle facultatives», à la page 215
- «Exemple ASNCLP de création d'une table cible CCD», à la page 216
- «Utilisation de tables CCD comme sources de réplication Q», à la page 216

Important pour la version 10.1 sous Linux, UNIX et Windows : Si la base de données source est DB2 10.1 for Linux, UNIX et Windows avec plusieurs membres DB2 pureScale, la réplication Q ne prend pas en charge des tables cible CCD. Avec la version 10.1 en tant que source, les cibles CCD sont uniquement prises en charge dans des bases de données à membre unique avec le paramètre **compatibility** de Q Capture défini à 0907 ou à une valeur inférieure.

Attributs Condensée et complète

Deux attributs définissent une table CCD : condensée et complète. La liste suivante offre un récapitulatif de ces attributs :

Complète (COMPLETE=Y)

Une table CCD complète contient chaque ligne provenant de la table source et ayant un intérêt et est initialisée avec un ensemble complet de données source.

Toutes les options de chargement de la table cible sont valides pour des tables CCD complètes (automatique, manuel ou pas de chargement).

Incomplète (COMPLETE=N)

Une table CCD incomplète contient uniquement les changements apportés à la table source et est initialisée sans données.

L'unique option de chargement valide pour des tables CCD incomplètes est l'option pas de chargement.

Condensée (CONDENSED=Y)

Une table CCD condensée contient une ligne pour chaque valeur clé de la table source et contient uniquement la valeur la plus récente de la ligne.

Pour les tables CCD condensées, une clé primaire est nécessaire pour garantir la non duplication des lignes. Dans le cas d'un conflit de mise à jour, toutes les colonnes source seront forcées dans la ligne. Les paramètres requis pour la règle de conflit et l'action de conflit sont CONFLICT_RULE=K et CONFLICT_ACTION=F.

Non condensée (CONDENSED=N)

Une table CCD non condensée contient plusieurs lignes avec la même valeur clé, une ligne pour chaque opération UPDATE, INSERT ou DELETE au niveau de la table source.

Lorsqu'elles sont ajoutées à la table CCD, toutes les lignes deviennent des opérations INSERT. Les tables CCD non condensées ne peuvent pas avoir un index unique ou une clé primaire.

Une table CCD utilisée pour conserver un historique des modifications apportées à la table source est complète (COMPLETE=Y) et non condensée (CONDENSED=N).

Options de chargement des tables CCD

Les options de chargement suivantes s'appliquent aux tables CCD cible :

Complète

Vous pouvez spécifier un chargement automatique, un chargement manuel ou aucun chargement du programme Q Apply.

Incomplète

Vous ne devez spécifier aucun chargement.

Options concernant les erreurs et les conflits

Les options suivantes sont disponibles pour la gestion de conditions inattendues dans les tables cible CCD :

Condensée et complète

Deux choix sont possibles pour la détection de conflits :

- Forcer l'application des modifications de la table source à la table cible (CONFLICT_ACTION=F).
- Ignorer la condition et poursuivre (CONFLICT_ACTION=I).

Pour tous les types de table CCD, la seule option de conflit valide consiste à contrôler uniquement les colonnes clés (CONFLICT_RULE=K).

Notes :

1. Pour les cibles CCD incomplètes, la table IBMQREP_EXCEPTIONS n'est pas mise à jour après "row not found" ou les conflits de lignes en double car les lignes de la table source peuvent ne pas exister dans une cible CCD incomplète et le reconditionnement peut se répéter fréquemment.
2. Pour les cibles CCD complètes, la table IBMQREP_EXCEPTIONS est mise à jour après les conflits sauf dans le cas suivant : les violations de ligne en double ne sont pas consignées car les lignes traitées par les opérations de suppression demeurent dans les cibles CCD et le reconditionnement peut se répéter fréquemment.
3. Pour les tables CCD condensées, la mise à jour de la clé de la table source peut résulter de la suppression d'une ligne sur la cible CCD. La ligne supprimée est généralement une ligne "delete" (dont la colonne IBMSNAP_OPERATION contient la valeur D). Ce type de ligne est supprimé uniquement s'il possède la même valeur de clé que la nouvelle valeur de la mise à jour de clé.
4. Pour les tables CCD condensées, CONFLICT_ACTION=I implique que les conflits survenant sur la cible sont ignorés sauf dans le cas suivant : si une mise à jour de clé sur la source implique un reconditionnement sur la cible CCD, celui-ci est nécessaire à la cohésion des données CCD.

Colonnes par défaut des tables CCD

Par définition, une table CCD comprend toujours les colonnes suivantes en plus des colonnes répliquées à partir de la table source. Ces colonnes contiennent des informations du journal de reprise sur la base de données source qui fournit des détails supplémentaires sur chaque modification de ligne :

Colonne	Description
IBMSNAP_INTENTSEQ	<p>Type de données : CHAR(10) POUR BIT DATA ; NULL admis : Non.</p> <p>Numéro de séquence qui identifie uniquement un changement. Cette valeur est croissante dans une transaction.</p> <p>z/OS Numéro de séquence du journal (LRSN ou RBA) de chaque opération de mise à jour, de suppression et d'insertion.</p>
IBMSNAP_OPERATION	<p>Type de données : CHAR(1); Null admis : Non</p> <p>Indicateur qui spécifie le type d'opération : I (INSERT), U (UPDATE) ou D (DELETE).</p>
IBMSNAP_COMMITSEQ	<p>Type de données : CHAR(10) POUR BIT DATA ; NULL admis : Non.</p> <p>Numéro de séquence pour chaque ligne au sein d'une transaction.</p> <p>z/OS Numéro de séquence du journal (LRSN ou RBA) de l'enregistrement de validation source.</p>
IBMSNAP_LOGMARKER	<p>Type de données : TIMESTAMP; Null admis : Non</p> <p>Heure approximative du serveur source dont les données ont été validées (temps moyen de Greenwich, GMT).</p> <p>Le paramètre logmarkertz de Q Apply permet de spécifier que l'horodatage utilise l'heure locale du serveur Q Apply. Spécifier le paramètre logmarkertz=local n'est utile que lorsque les serveurs Q Capture et Q Apply sont situés dans le même fuseau horaire avec une heure d'été (ou autre) identique.</p>

Colonnes de contrôle facultatives

Lorsque vous créez une table CCD incomplète (COMPLETE=N) à l'aide du programme de ligne de commande ASNCLP ou du Centre de réplication, vous pouvez spécifier des colonnes de contrôle facultatives. Le tableau suivant décrit ces colonnes :

Colonne	Description
IBMSNAP_AUTHID	<p>Type de données : VARCHAR(30); Null admis : Oui</p> <p>ID utilisateur qui a mis à jour la table source.</p> <p>z/OS Cette colonne correspond à l'ID utilisateur primaire.</p>
IBMSNAP_AUTHTKN	<p>Type de données : VARCHAR(30); Null admis : Oui</p> <p>Clé d'accès qui est associée à la transaction.</p> <p>z/OS ID de corrélation (normalement un nom de travail) qui a exécuté la mise à jour source.</p>
IBMSNAP_PLANID	<p>Type de données : VARCHAR(8); Null admis : Oui</p> <p>z/OS Nom de plan associé à la transaction. Cette colonne doit être NULL pour DB2 for Linux, UNIX et Windows.</p>
IBMSNAP_UOWID	<p>Type de données : CHAR(10) FOR BIT DATA; Null admis : Oui</p> <p>Identificateur de l'unité de travail (UOW) provenant de l'enregistrement de journal pour une ligne.</p> <p>z/OS Identificateur de l'unité de travail, parfois appelé ID unité de récupération (URID) de la transaction.</p>

Exemple ASNCLP de création d'une table cible CCD

Vous pouvez utiliser le Centre de réplication ou le programme de ligne de commande ASNCLP pour créer des tables CCD dans le cadre du processus de création d'un abonnement Q pour la table source. L'exemple ASNCLP suivant définit l'environnement, puis crée un abonnement Q pour la table EMPLOYEE dans la base de données SAMPLE, en créant également une table cible CCD :

```
ASNCLP SESSION SET TO Q REPLICATION;
SET SERVER CAPTURE TO DB SAMPLE;
SET SERVER TARGET TO DB TARGET;
CREATE QSUB USING REPLQMAP REPQMAP1
(SUBNAME SUB1 EMPLOYEE TYPE CCD);
```

Utilisation de tables CCD comme sources de réplication Q

Vous pouvez utiliser une table cible CCD remplie par Q Apply comme table source pour une autre base de données cible de réplication Q. En règle générale, cette configuration utilise trois niveaux, dans lesquels les valeurs d'enregistrement du journal émanant de la base de données source au niveau 1 sont envoyées au programme Q Apply au niveau 2 et appliquées à la table CCD cible. La table CCD au niveau 2 est alors mappée vers une table CCD au niveau 3, et les valeurs d'origine du niveau 1 sont alors propagées au niveau 3.

Pour définir une configuration à trois niveaux similaire, utilisez le programme ASNCLP pour créer un abonnement Q du niveau 1 vers le niveau 2. Dans la commande CREATE QSUB, utilisez les mots-clés TYPE CCD pour indiquer que la table CCD cible au niveau 2 sera remplie directement avec des valeurs extraites directement du journal de reprise au niveau 1. Ensuite, lorsque vous créez l'abonnement Q du niveau 2 vers le niveau 3, remplacez les mots-clés TYPE CCD par TYPE USERTABLE dans la commande CREATE QSUB, de sorte que les colonnes CCD au niveau 2 soient mappées vers les colonnes correspondantes au niveau 3.

Les exemples de commande CREATE QSUB suivants illustrent cette méthode. Le premier exemple utilise les mots-clés TYPE CCD pour créer un abonnement Q du niveau 1 vers le niveau 2 :

```
CREATE QSUB USING TIER1_ASN_TO_TIER2_ASN
(SUBNAME EMPLOYEE0001 HR.EMPLOYEE
 TARGET NAME HR.TIER2EMPLOYEE
 TYPE CCD
 KEYS (C1));
```

Le second exemple omet les mots-clés TYPE CCD et utilise TYPE USERTABLE à la place. Cette syntaxe invite le programme Q Apply à créer une table avec des colonnes identiques au niveau 3 et à propager les valeurs de la table CCD au niveau 2 vers les colonnes correspondantes au niveau 3 :

```
CREATE QSUB USING TIER2_ASN_TO_TIER3_ASN
(SUBNAME EMPLOYEE0001 HR.TIER2EMPLOYEE
 TARGET NAME HR.TIER3EMPLOYEE
 TYPE USERTABLE
 KEYS (C1));
```

L'utilisation d'une table de réplication Q en tant que source requiert la version 9.7 avec groupe de correctifs 5 ou ultérieure sous Linux, UNIX et Windows, et la version 10.1 sous z/OS avec PTF correspondant au groupe de correctifs 5.

Chapitre 12. Création de publications

Avec la publication d'événements, vous pouvez publier les enregistrements mis à jour ou les mises à jour provenant d'une table source vers une application utilisateur. Le programme Q de capture publie les mises à jour issues d'une table source et place ces mises à jour dans une file d'attente pour envoi. Il vous revient ensuite de définir l'application qui va réceptionner ces mises à jour.

Conseil : Le programme d'essai asnqwxml fournit un exemple d'application Web qui consomme les messages XML publiés par le programme Q Capture. L'exemple indique comment utiliser les publications dans un scénario métier.

Regroupement des publications et des mappes de files d'attente pour la publication

Avant de définir des publications XML et des mappes de files d'attente pour la publication, vous devez planifier leur mode de regroupement.

Chaque publication XML identifie une table source unique à partir de laquelle les modifications seront publiées au format XML ou dans un format délimité, tel que des valeurs séparées par des virgules (CSV). Lorsque vous définissez une publication XML, vous devez également définir la mappe de files d'attente pour la publication qui est utilisée pour acheminer les données de cette table source. Entre autres, chaque mappe de files d'attente pour la publication identifie la file WebSphere MQ à laquelle le programme Q Capture envoie des modifications. Une seule mappe de files d'attente pour la publication peut être utilisée pour acheminer les données : vous pouvez donc sélectionner les publications qui utiliseront la même mappe pour acheminer les données.

Lorsque vous planifiez le regroupement de publications XML et de mappes de files d'attente pour la publication, gardez à l'esprit les règles suivantes :

- Une file d'attente WebSphere MQ ne peut pas être partagée par plusieurs programmes Q Capture.
- Un programme Q Capture unique peut écrire sur plusieurs files d'attente d'envoi.
- Vous pouvez créer une ou plusieurs mappes de files d'attente pour la publication à partir d'un seul programme Q Capture.

Fonctionnement du programme Q Capture avec la file d'attente d'envoi

Pour une mappe de files de publication, le programme Q Capture capture des modifications dans le journal de base de données pour toutes les tables pour lesquelles il existe des publications XML actives. Q Capture stocke en mémoire ces modifications jusqu'à ce qu'il lise l'entrée de validation ou d'abandon correspondante du journal de base de données. Le programme Q Capture envoie alors les informations concernant les transactions validées à toutes les files d'attente d'envoi WebSphere MQ qui ont été définies pour les publications XML.

Suggestions pour le regroupement de publications XML similaires avec des mappes de files d'attente pour la publication

Pour des tables impliquées dans des transactions avec une ou plusieurs applications, vous devez créer des publications XML pour ces tables afin qu'elles partagent toutes une même mappe de files pour la publication. Le regroupement de publications XML similaires avec la même mappe de files pour la publication assure la cohérence transactionnelle des données envoyées à la file d'attente d'envoi.

Il est important que les publications XML comportant des dépendances partagent la même mappe de files pour la publication. Si vous définissez des publications XML impliquées dans des transactions associées permettant d'envoyer des données par l'intermédiaire de mappes de files indépendantes, le programme Q Capture partage les données entre les diverses files d'attente d'envoi.

Si plusieurs applications mettent à jour le serveur source mais pas les mêmes tables et que vous configurez un programme Q Capture unique pour publier des données du serveur source vers un serveur cible, vous pouvez envisager de définir plusieurs mappes de files d'attente pour la publication à l'usage de ce programme Q Capture. Toutes les publications XML qui sont associées dans les transactions pour chaque application sont alors publiées sur l'une de ces mappes de files. Une telle configuration peut avoir des avantages, comme l'isolement des incidents ou l'accroissement du rendement. Il est possible d'accroître le rendement et d'améliorer l'isolement en cas d'incident en configurant chaque programme Q Capture avec sa propre mappe de files pour la publication. Cependant, ces avantages sont contrebalancés par une consommation plus importante de l'unité centrale et un environnement de publication plus complexe.

Création de mappes de files pour la publication

Lorsque vous créez des publications, vous indiquez à quelle file d'attente WebSphere MQ envoyer les données en associant chaque publication XML à une mappe de files pour la publication. Vous pouvez créer une mappe de files pour la publication avant de commencer à créer des publications ou au cours de la procédure de création.

Avant de commencer

- Planifiez le regroupement des publications et des mappes de files pour la publication.
- Sur le serveur hébergeant les tables source des publications, créez les tables de contrôle pour le programme Q Capture.
- Assurez-vous que vous avez défini les objets appropriés dans WebSphere MQ.

Restrictions

La même file d'attente d'envoi ne peut pas être utilisée à la fois pour une réplication Q et une publication d'événements car une file d'attente d'envoi peut transférer des messages condensés (pour une réplication Q) ou des messages XML (pour la publication d'événements), mais pas les deux.

Procédure

Pour créer une mappe de files pour la publication, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande CREATE PUBQMAP. Par exemple, les commandes suivantes définissent l'environnement et créent une mappe de files pour la publication SAMPLE_EP1_TO_SUBSCRIBER : <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE EP1; SET RUN SCRIPT LATER; CREATE PUBQMAP SAMPLE_EP1_TO_SUBSCRIBER USING SENDQ "EP1.QM1.PUBDATAQ" MESSAGE CONTENT TYPE R MAX MESSAGE SIZE 128 HEARTBEAT INTERVAL 5;</pre>
Centre de réplication	Utilisez la fenêtre Création d'une mappe de files d'attente pour la publication. Pour ouvrir cette fenêtre, développez le schéma Q Capture qui identifie le programme Q Capture utilisant la mappe de files. Cliquez avec le bouton droit sur le dossier Mappes de files pour la publication et sélectionnez Création .

Conseil : Vous pouvez utiliser l'un des outils d'administration de la réplication pour valider la file d'attente d'envoi spécifiée pour une mappe de files pour la publication. Cliquez sur **Valider les files d'attente** dans la fenêtre Création d'une mappe de files d'attente pour la publication ou utilisez la commande **VALIDATE WSMQ ENVIRONMENT FOR** du programme ASNCLP.

Lorsque vous créez une mappe de files d'attente pour la publication, vous définissez les options suivantes :

File d'attente d'envoi

File WebSphere MQ à laquelle le programme Q Capture envoie des données source et des messages d'informations.

Contenu des messages

Vous pouvez déterminer que le programme Q Capture envoie des messages avec l'un des types suivants de contenu :

- Opérations sur des lignes individuelles (Ce type de message du programme Q Capture est appelé *message d'opération sur la ligne*).
- Transactions complètes (Ce type de message du programme Q Capture est appelé *message de transaction*).

Pour les deux types de contenu, l'opération n'est envoyée que lorsque la transaction concernée a été validée. Le type de contenu des messages que vous choisissez détermine le mode d'envoi des données par le programme Q Capture pour toutes les publications utilisant cette mappe de files pour la publication.

Pour les types de données LOB : Quelle que soit l'option choisie, les types de données LOB sont envoyés séparément sous forme de messages physiques individuels associés au message de transaction ou au message d'opération de ligne.

Longueur maximale du message

Taille maximale d'un message (en kilooctets) que le programme Q Capture peut placer dans sa file d'attente d'envoi. Cette longueur maximale doit être inférieure ou égale à l'attribut de taille maximale des messages de WebSphere MQ (MAXMSGL) qui est défini pour la file d'attente ou le gestionnaire de files d'attente.

Action d'erreur de file d'attente

Actions effectuées par le programme Q Capture lorsque la file d'attente d'envoi n'accepte plus de messages en raison d'une erreur (lorsque la file d'attente est saturée par exemple) :

- Arrêt du fonctionnement
- Arrête de placer des messages dans la file d'attente en erreur mais continue à placer des messages dans les autres files d'attente

Intervalle des pulsations

Fréquence (en secondes) avec laquelle le programme Q Capture envoie des messages à cette file d'attente pour indiquer à l'application utilisateur qu'il est toujours en cours d'exécution lorsqu'il n'y a aucune modification à publier. Le message de pulsation est envoyé au premier intervalle de validation après l'expiration de l'intervalle des pulsations. Une valeur égale à 0 indique au programme Q Capture de ne pas envoyer de messages de pulsation.

Remarque : Cet intervalle des pulsations est différent du paramètre HBINT de WebSphere MQ que vous pouvez définir pour un canal WebSphere MQ.

Erreurs de conversion de pages de codes

Si les données de type caractères sont publiées lorsque les données causent une erreur de conversion de page de code. Par défaut, aucune donnée n'est envoyée pour la zone de caractère qui n'a pas réussi la conversion de la page de code pour les messages XML. Pour le format de message délimité, par défaut aucune des données des colonnes de caractères dans la ligne n'est envoyée lorsqu'une zone à caractère unique échoue dans la conversion de la page de code. A la place, toutes les colonnes de caractères sont envoyées en tant que valeurs null. Cochez la case pour spécifier les valeurs hexadécimales à envoyer à la place des valeurs null.

En-tête de message

Indique s'il convient ou non d'inclure un en-tête compatible JMS (MQRFH2) dans tous les messages placés dans la file d'attente d'envoi. Lorsque vous créez une publication qui emprunte la file d'attente d'envoi, vous pouvez indiquer une rubrique pour la zone de rubrique de l'en-tête. Une rubrique est une chaîne de caractères qui décrit la nature des données publiées.

Format délimité

Si vous voulez ou non que le programme Q Capture publie les messages en format délimité. Vous pouvez accepter les délimiteurs par défaut pour la colonne, les données de caractère, les nouvelles lignes, et les décimales, ou utiliser les quatre zones pour spécifier vos propres délimiteurs.

Si vous spécifiez vos propres délimiteurs, chacune des zones de délimiteur doit avoir une valeur différente. Vous pouvez faire un choix dans la liste déroulante de délimiteurs valides, entrer un caractère ou saisir un point de code caractères. Dans un point de code caractères, vous utilisez la syntaxe 0xJJ ou XJJ où JJ est la représentation hexadécimale du point de code. Par exemple, pour le délimiteur #, vous pouvez utiliser #, 0x23 ou X23.

Page de codes pour la publication de messages délimités

Affiche la page de codes que le programme Q Capture utilise pour publier des messages délimités. Si cette page de codes est différente de la page de codes de la table source, le programme Q Capture convertira les données. Pour éviter la conversion de page de codes, indiquez la page de codes utilisée pour la table source.

Délimiteur de colonne

Spécifie un délimiteur pour séparer les valeurs dans chaque colonne de la table source. Le délimiteur par défaut est une virgule (,). Une valeur de colonne Null est représentée par deux délimiteurs de colonnes consécutifs.

Délimiteur de chaîne de caractères

Spécifie un délimiteur comprenant toutes les données de types caractères. Le délimiteur par défaut est un double guillemets (""). Si les données de la chaîne de type caractères contiennent leur propre délimiteur, ce dernier est précédé en lui ajoutant le même délimiteur comme préfixe.

Nouvelle valeur de ligne

Spécifie un délimiteur de nouvelle ligne pour séparer les enregistrements de données de changement dans un message. La valeur par défaut est un saut de ligne (LF).

Valeur décimale

Spécifie un caractère à utiliser pour la virgule décimale. La valeur par défaut est un point (.).

Création de publications

La fonction de publication d'événements permet de publier des lignes ou des transactions modifiées à partir d'une table source dans une application utilisateur, par la création de publications.

Chaque publication est un objet unique qui identifie les éléments suivants :

- la table source à partir de laquelle vous souhaitez publier des modifications ;
- les colonnes et lignes de la table source que vous souhaitez publier ;
- la mappe de files pour la publication, qui nomme la file d'attente WebSphere MQ sur laquelle les modifications sont publiées.

En publication d'événements, vous pouvez créer une ou plusieurs publications à la fois.

Avertissement : Les publications sont des objets distincts des abonnements Q. Les publications ne publient pas de données dans le programme Q Apply mais vers une application de votre choix. Elles servent à publier des données, tandis que les abonnements Q permettent de les répliquer. Si vous souhaitez répliquer des modifications d'une table source et demandez au programme Q Apply de les appliquer à une table cible ou de les transmettre à une procédure stockée pour procéder à la manipulation de données, créez un abonnement Q, pas une publication.

Les rubriques suivantes expliquent comment créer une ou plusieurs publications et les personnaliser en fonction de vos besoins.

Création de publications

Lorsque vous créez une publication XML, vous définissez la publication des données sous forme de messages XML ou délimités à partir d'une table source vers WebSphere MQ afin de permettre à une application d'abonnement d'extraire et d'utiliser ces messages.

Avant de commencer

- Planifiez le regroupement des publications et des mappes de files pour la publication.
- Sur le serveur qui héberge la table source de la publication XML, créez les tables de contrôle pour le programme Q Capture.
- Créez une mappe de files d'attente pour la publication. (Vous pouvez effectuer cette tâche avant ou pendant la création d'une publication XML.)

Restrictions

- Une vue ne peut pas constituer une source de publication XML.

A propos de cette tâche

La figure 26 indique comment une publication XML unique connecte une table source à une file d'attente d'envoi WebSphere MQ.

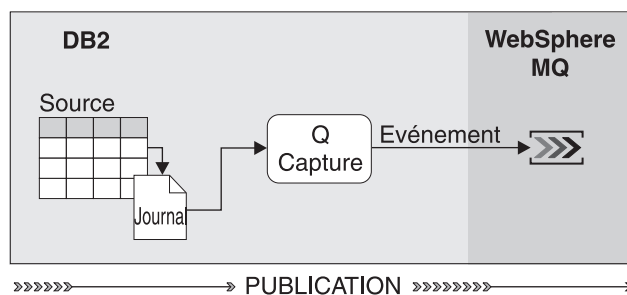


Figure 26. *Publication unique.* Les modifications provenant d'une table source sont publiées par le biais des files d'attente WebSphere MQ.

La figure 27, à la page 225 indique comment plusieurs publications XML peuvent utiliser la même mappe de files pour la publication et le même programme Q Capture.

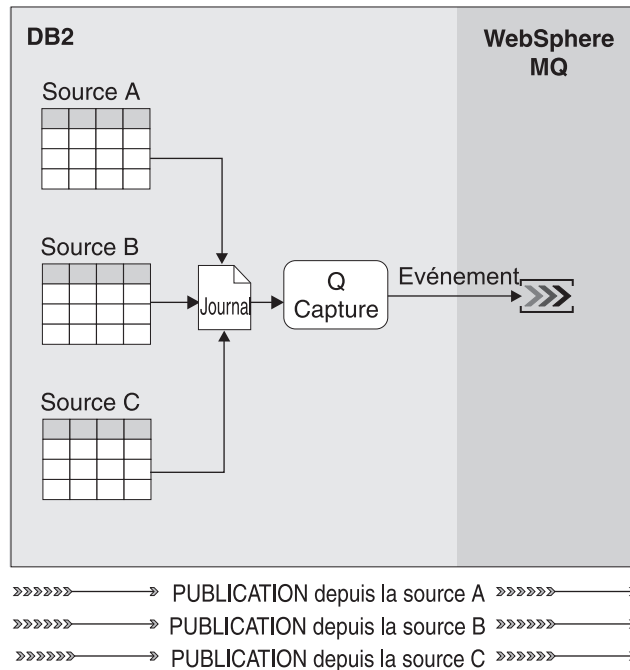


Figure 27. Publications multiples. Les modifications provenant des tables source sont publiées par le biais des files d'attente WebSphere MQ.

Procédure

Pour créer des publications, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande CREATE PUB . Par exemple, les commandes suivantes définissent l'environnement et crée la publication DEPARTMENT0001 : <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE EP1; SET RUN SCRIPT LATER;</pre> <pre>CREATE PUB USING PUBQMAP SAMPLE_EP1_TO_SUBSCRIBER (PUBNAME "DEPARTMENT0001" JK.DEPARTMENT ALL CHANGED ROWS Y SUPPRESS DELETES Y);</pre>
Centre de réplication	Utilisez l'assistant Création de publications. Pour ouvrir cet assistant, développez le schéma Q Capture qui identifie le programme Q Capture qui doit capturer les modifications pour la publication XML. Cliquez avec le bouton droit de la souris sur le dossier Publications XML et sélectionnez Création . Vous pouvez créer une ou plusieurs publications XML à l'aide de l'assistant. Lorsque vous créez plusieurs publications XML à la fois, le Centre de réplication part du principe que vous souhaitez répliquer toutes les colonnes et lignes de chaque table source. À la fin de l'assistant, avant que le Centre de réplication ne génère les publications XML, vous pouvez modifier certaines d'entre elles afin de ne publier qu'un sous-ensemble des colonnes et lignes source.

Colonnes source pour les publications XML

Par défaut, lorsque vous créez des publications XML, les modifications apportées à toutes les colonnes de la table source sont publiées. Cependant, vous pouvez publier un sous-ensemble des colonnes si vous ne souhaitez pas rendre toutes les colonnes de la table source accessibles à l'application utilisateur.

Vous pouvez également publier un sous-ensemble des colonnes si l'application utilisateur d'une publication ne prend pas en charge tous les types de données définis pour la table source.

Pour publier un sous-ensemble des colonnes, sélectionnez uniquement les colonnes source que vous souhaitez publier dans l'application utilisateur. Si vous créez une publication unique, l'assistant Création de publications du Centre de réplication vous propose des options permettant de publier un sous-ensemble des colonnes de la table source. Si vous créez plusieurs publications à la fois, sur la page Vérification de l'assistant Création de publications, sélectionnez la publication pour laquelle vous ne souhaitez publier qu'un sous-ensemble des colonnes, puis modifiez les propriétés de cette publication.

Particularité des colonnes d'objet LOB : Si vous sélectionnez des colonnes contenant des types de données LOB pour une publication, assurez-vous que la table source respecte au moins une contrainte de base de données unique (par exemple, un index ou une clé primaire unique). Il n'est pas nécessaire de sélectionner les colonnes qui composent cette propriété d'unicité pour la publication.

Publication par le programme Q Capture d'un message pour les publications

Lorsque vous créez des publications, vous pouvez demander au programme Q Capture de publier un message chaque fois qu'une colonne de la table source change ou uniquement lorsque les colonnes appartenant à une publication changent.

Les sections suivantes décrivent les deux types d'événements pouvant entraîner la publication d'un message par le programme Q Capture :

- «Message envoyé uniquement en cas de modification des colonnes de publication»
- «Message envoyé à chaque modification de la table source», à la page 227

Recommandation : En général, l'option appropriée consiste à ne publier que les modifications portant sur une colonne sélectionnée. Cependant, certaines applications n'ont besoin que d'une partie d'une ligne, par exemple les colonnes de clé, lorsqu'une modification est effectuée. Ces informations publiées peuvent servir de notification d'événement, qui permettent de déclencher l'exécution d'autres opérations. Si vous publiez toutes les colonnes contenues dans la table source, ces deux options entraînent la même opération.

Message envoyé uniquement en cas de modification des colonnes de publication

Par défaut, le programme Q Capture ne publie un message que lorsque la modification porte sur les colonnes que vous avez sélectionnées pour les publications.

Exemple : supposons que votre table source comporte 100 colonnes et que vous en sélectionnez 25 à publier dans une publication. Si vous demandez l'envoi d'un message uniquement en cas de modification des colonnes de publication, le programme publie un message à chaque modification d'une des 25 colonnes de cette publication. En revanche, il n'en publie pas lorsqu'une modification est apportée à l'une des 75 colonnes n'appartenant pas à la publication.

Message envoyé à chaque modification de la table source

Vous pouvez définir des publications de sorte que le programme Q Capture publie un message chaque fois qu'une modification a lieu dans la table source. Si vous ne publiez qu'un sous-ensemble des colonnes de la table source, le programme Q Capture publie un message, même si la modification est effectuée une colonne n'appartenant pas à une publication.

Exemple : supposons que votre table source comporte 100 colonnes et que vous en sélectionnez 25 à publier dans une publication. Si vous demandez l'envoi d'un message à chaque modification d'une table source, le programme Q Capture publie un message à chaque modification d'une des 100 colonnes de la table source.

Conditions de recherche pour filtrer les lignes dans les publications XML

Par défaut, lorsque vous créez des publications XML, toutes les lignes de la table source sont publiées. Cependant, lorsque vous créez une publication, vous pouvez spécifier une clause WHERE comportant une condition de recherche pour identifier les lignes que vous souhaitez publier.

Lorsque le programme Q Capture détecte une modification du journal de récupération DB2 qui est associé à une table source, il compare cette modification à la condition de recherche pour déterminer s'il convient ou non de la publier.

Si vous créez une publication unique, l'assistant Création de publications du Centre de réplication vous permet d'ajouter une clause WHERE afin de publier un sous-ensemble des lignes de la table source. Si vous créez plusieurs publications XML à la fois, sur la page Vérification de l'assistant Création de publications, sélectionnez la publication pour laquelle vous souhaitez définir un sous-ensemble des lignes, puis modifiez les propriétés de cette publication afin d'ajouter la clause WHERE.

Lorsque vous spécifiez une clause WHERE, vous pouvez indiquer si la colonne est évaluée avec des valeurs de l'enregistrement de journal en cours. Si vous souhaitez qu'une colonne soumise à la clause WHERE soit évaluée avec des valeurs de cet enregistrement, mettez deux points devant le nom de la colonne.

Exemple de clause WHERE qui évalue une colonne avec des valeurs de l'enregistrement de journal en cours

```
WHERE :LOCATION = 'EAST' AND :SALES > 100000
```

Dans cet exemple, LOCATION et SALES sont les noms de colonnes de la table source qui sont évaluées. Le programme Q Capture envoie ici uniquement les changements de la table source qui concernent les ventes à l'Est supérieures à 100 000\$. Lorsque vous entrez un nom de colonne, les caractères sont en majuscule sauf si mettez le nom entre guillemets. Par exemple, saisissez "Location" si le nom de la colonne doit respecter la distinction de la casse.

Si le programme Q Capture publie une colonne faisant partie de la clause WHERE, il devra peut-être modifier le type d'opération à envoyer à la table ou procédure stockée cible.

Exemple dans lequel le programme Q Capture doit modifier le type d'opération en raison d'une clause WHERE

```
WHERE :LOCATION = 'EAST'  
AND :SALES > 100000
```

Supposons que la modification suivante soit apportée à la table source :

```
INSERT VALUES ( 'EAST', 50000 )  
UPDATE SET SALES = 200000 WHERE LOCATION = 'EAST'
```

La valeur antérieure ne répondant pas à la condition de recherche de la clause, Q Capture envoie l'opération en tant que INSERT au lieu de UPDATE.

De même, si la valeur antérieure répond à la condition de recherche mais pas la valeur après, Q Capture envoie alors les changements UPDATE vers DELETE. Par exemple, si vous avez la même clause WHERE que précédemment :

```
WHERE :LOCATION = 'EAST'  
AND :SALES > 100000
```

Supposons maintenant que la modification suivante soit apportée à la table source :

```
INSERT VALUES ( 'EAST', 200000 )  
UPDATE SET SALES = 50000 WHERE LOCATION = 'EAST'
```

La première modification, l'insertion, est envoyée vers la table cible ou la procédure stockée, car il répond à la condition de recherche de la clause WHERE (200000 > 100000 est vérifié). Cependant, la deuxième modification, la mise à jour, ne répond pas à la condition de recherche (50000 > 100000 n'est pas vérifié). Q Capture envoie le changement en tant que DELETE afin que la valeur soit supprimée de la table cible ou de la procédure stockée.

Conditions de recherche complexes

La publication d'événements permet de définir des clauses WHERE plus complexes. Cependant, les conditions de recherche complexes peuvent avoir une incidence sur les performances. Vous pouvez par exemple spécifier une clause plus complexe avec une sous-requête qui référence d'autres tables ou enregistrements de la table source ou d'une autre table.

Exemple de clause WHERE avec sous-requête

```
WHERE :LOCATION = 'EAST'  
AND :SALES > (SELECT SUM(EXPENSE) FROM STORES WHERE STORES.DEPTNO = :DEPTNO)
```

Dans cet exemple, Q Capture envoie uniquement les changements effectués à l'Est ayant généré des bénéfices, où la valeur de la vente est supérieure au montant total des dépenses. La sous-requête référence la table STORES et les colonnes suivantes de la table source : LOCATION, SALES et DEPTNO.

Lorsque vous définissez une publication avec une sous-requête dans une clause WHERE, les problèmes suivants risquent de se produire :

- Les performances peuvent être ralenties car, pour chaque changement dans la table source, Q Capture calcule une large sélection de la table STORES pour calculer la valeur SUM(EXPENSE). Ce type de sélection peut également intervenir sur les verrouillages des tables.
- La sous-requête peut produire des résultats inattendus. Par exemple, la sous-requête étant évaluée par rapport aux valeurs de la base de données en cours, l'exemple ci-dessus donne une réponse fautive si la valeur EXPENSE change dans la base de données, tandis que les colonnes de la clause WHERE sont remplacées par les valeurs précédentes de l'enregistrement de journal. Si le nom de la table que la sous-requête référence ne change pas, la condition de recherche fournit alors les bons résultats.

Restrictions des conditions de recherche

- Les conditions de recherche ne peuvent pas contenir de fonctions de colonne, à moins que la fonction figure dans une instruction de sous-requête.

Clause WHERE non valide avec pour fonctions de colonne :

```
#-----
# Incorrect : à éviter
#-----
```

```
WHERE :LOCATION = 'EAST' AND SUM(:SALES) > 1000000
```

Le Centre de réplication valide les conditions de recherche lorsque le programme Q Capture les évalue, pas lorsque le Centre de réplication crée la publication. Si une publication contient une condition de recherche non valide, cette publication échoue lorsque la condition non valide est évaluée, et les publications sont désactivées.

- Les conditions de recherche ne peuvent contenir une clause ORDER BY ou GROUP BY à moins que cette clause ne soit dans une instruction de sous-requête.

Clause WHERE non valide avec GROUP BY :

```
#-----
# Incorrect : à éviter
#-----
```

```
WHERE :COL1 > 3 GROUP BY COL1, COL2
```

Clause WHERE valide avec GROUP BY :

```
WHERE :COL2 = (SELECT COL2 FROM T2 WHERE COL1=1 GROUP BY COL1, COL2)
```

- Les conditions de recherche ne peuvent pas référencer le nom réel de la table source dont vous publiez les modifications. N'utilisez pas la notation schema.tablename dans une clause WHERE pour le nom en cours de la table source. Cependant, vous pouvez référencer un autre nom de table dans une sous-requête en utilisant une notation schema.tablename.

Clause WHERE non valide avec pour nom en cours de la table source et nom de colonne :

```
#-----
# Incorrect : à éviter
#-----
```

```
WHERE :ADMINISTRATOR.SALES > 100000
```

Dans cet exemple de clause WHERE comportant les noms réels de la table source et des colonnes, la table publiée est ADMINISTRATOR et le nom de colonne, SALES. Cette clause WHERE non valide ne vise à sélectionner que les valeurs de la colonne SALES de la table ADMINISTRATOR, pour lesquelles la valeur de SALES est supérieure à 100000.

Clause WHERE valide avec pour nom de colonne :

```
WHERE :SALES > 100000
```

Dans cet exemple de clause WHERE comportant un nom de colonne, SALES est le nom de la colonne.

- Les conditions de recherche ne peuvent pas référencer des valeurs qui figuraient dans les colonnes avant qu'une modification soit effectuée ; elles ne peuvent renvoyer qu'à des valeurs après.
- Les conditions de recherche ne peuvent contenir les prédicats EXISTS.
- Elles ne peuvent contenir un prédicat quantifié, qui est un prédicat utilisant SOME, ANY ou ALL.
- Elles ne peuvent pas non plus référencer les valeurs objet LOB.

Colonnes de clé pour les publications XML

Pour chaque publication XML, vous devez indiquer les colonnes de la table source qui sont des colonnes de clé. La publication d'événements exige des colonnes de clé afin de respecter la règle selon laquelle chaque ligne doit être unique. Vous pouvez soit demander au Centre de réplication de recommander les colonnes de la table source à utiliser pour garantir l'unicité des lignes, soit sélectionner vous-même les colonnes de clé.

Si vous créez une publication unique, l'assistant Création de publications du Centre de réplication lance la fenêtre Sélection de colonnes de clé, qui permet de sélectionner les colonnes de clé dans la table source. Si vous créez plusieurs publications à la fois, vous pouvez utiliser la page Vérification de l'assistant Création de publications pour personnaliser les colonnes de clé à utiliser.

Options pour l'intégration de colonnes non modifiées dans les messages pour les publications XML

Lorsque vous créez des publications qui publient un sous-ensemble des colonnes source, vous pouvez indiquer les valeurs de colonne de chaque ligne que le programme Q Capture inclut dans le message publié.

Les sections suivantes décrivent les valeurs que le programme Q Capture peut inclure dans le message :

- «Seules les colonnes modifiées sont envoyées»
- «Les colonnes modifiées et non modifiées sont envoyées», à la page 231

Cette option ne s'applique qu'aux valeurs contenues dans les colonnes sans clé. Le programme Q Capture publie toujours les valeurs dans des colonnes de clé.

Seules les colonnes modifiées sont envoyées

Par défaut, lorsque vous créez des publications, le programme Q Capture envoie les valeurs des colonnes que vous avez sélectionnées pour les publications en cas de modification de ces valeurs.

Exemple : supposons que votre table source comporte 100 colonnes et que vous en sélectionnez 25 à publier dans une publication. Si vous indiquez que seules les colonnes modifiées doivent être envoyées, à chaque modification de l'une des 25 colonnes sélectionnées, le programme Q Capture ne publie que les colonnes modifiées. Par exemple, si des modifications ont lieu dans 17 des 25 colonnes sélectionnées, le programme Q Capture envoie ces 17 valeurs modifiées.

Recommandation : Utilisez cette option pour réduire la quantité de données inutiles qui transitent par les files d'attente.

Les colonnes modifiées et non modifiées sont envoyées

Vous pouvez aussi définir des publications, de sorte que le programme Q Capture envoie toujours les valeurs des colonnes que vous avez sélectionnées pour les publications, que ces valeurs aient changé ou non.

Exemple : supposons que votre table source comporte 100 colonnes et que vous en sélectionnez 25 à publier dans une publication. Si vous indiquez que les colonnes modifiées et non modifiées doivent être envoyées, à chaque modification de l'une des 25 colonnes sélectionnées, le programme Q Capture publie toutes les colonnes sélectionnées. Par exemple, si des modifications ont lieu dans 17 des 25 colonnes sélectionnées, le programme Q Capture envoie quand même les valeurs des 25 colonnes.

Options pour l'intégration de valeurs avant dans les messages pour les publications

Lorsqu'une mise à jour a lieu dans des colonnes qui ne font pas partie de la clé cible, le programme Q Capture envoie la valeur de la colonne après la modification, ou envoie à la fois les valeurs avant et après la modification. Lorsqu'une mise à jour a lieu dans une colonne de clé, la valeur avant et la valeur après sont toujours envoyées.

Comme une opération de suppression s'applique toujours à une ligne et pas à une valeur de colonne particulière, les suppressions sont traitées différemment. Pour les suppressions, seules des valeurs avant sont envoyées. Les valeurs avant des colonnes de clé sont toujours envoyées. Si vous indiquez que le message doit inclure des valeurs de colonne antérieures et postérieures à la modification, si des valeurs de colonnes sans clé sont ensuite supprimées, les valeurs avant des colonnes sans clé sont envoyées.

Les sections ci-après décrivent les deux options pour les valeurs avant et après :

- «Envoi de nouvelles valeurs de données uniquement»
- «Envoi des valeurs de données anciennes et nouvelles»

Envoi de nouvelles valeurs de données uniquement

Par défaut, lorsqu'une mise à jour se produit dans la table source, le programme Q Capture publie les valeurs qui se trouvent dans les colonnes sans clé après la modification. Si vous indiquez que le message doit inclure uniquement les nouvelles valeurs de données (valeurs postérieures à la modification), le message n'inclut pas les valeurs qui figuraient dans les colonnes sans clé avant la modification.

Recommandation : Si l'application qui reçoit les messages pour les publications n'utilise jamais la valeur qui figurait dans chaque colonne sans clé avant la modification, indiquez que le programme Q Capture ne doit envoyer que les valeurs de colonnes postérieures à la modification.

Envoi des valeurs de données anciennes et nouvelles

Si vous indiquez que le message doit inclure à la fois les valeurs de données anciennes et nouvelles (valeurs antérieures et postérieures à la modification),

lorsqu'une colonne sans clé est mise à jour, le programme Q Capture publie la valeur qui figure dans la colonne avant la modification, et celle qui s'y trouve après la modification.

Recommandation : Si l'application qui reçoit les messages pour les publications utilise la valeur qui figurait dans chaque colonne avant la modification, indiquez que le programme Q Capture doit envoyer les valeurs de colonne antérieures et postérieures à la modification.

Restrictions pour les types de données LOB : Les valeurs avant des colonnes comportant des types de données LOB ne sont pas envoyées dans les messages. Si vous indiquez que le message comprenne à la fois les valeurs avant et les valeurs après, cette option ne s'applique pas aux colonnes comportant des types de données LOB et leurs valeurs avant ne sont pas envoyées.

Chapitre 13. Considérations relatives au type de données

Lorsque vous répliquez ou publiez certains types de données, telles que les données de type LONG VARCHAR ou LOB, vous devez prendre en compte certaines conditions et restrictions.

Restrictions générales de données pour la réplication Q et la publication d'événements

Certains types de données ne sont pas pris en charge dans la réplication Q et la publication d'événements et d'autres peuvent être utilisés seulement sous certaines conditions.

Restrictions sur le chiffrement des données

Vous pouvez répliquer ou publier certains types de données chiffrées :

z/OS EDITPROC

Les tables source DB2 for z/OS définies avec une routine de modification (EDITPROC) afin d'augmenter la sécurité des données sont prises en charge. Pour pouvoir utiliser ces tables comme source, le sous-système DB2 contenant les tables doit être un sous-système version 8 ou une version ultérieure, avec le correctif APAR PK13542 ou un niveau de correctif supérieur.

Linux UNIX Windows Fonction scalaire Encrypt

Les données de colonne peuvent être chiffrées et déchiffrées en utilisant la fonction de chiffrement scalaire dans DB2 pour Linux, UNIX et Windows. Pour l'application d'une réplication ou publication, les données doivent être de type VARCHAR FOR BIT DATA à la source. Ces données répliquent avec succès tant que la source et la cible utilisent la même page de code et que les fonctions de déchiffrement sont disponibles. La réplication des colonnes avec des données chiffrées ne doit être utilisée qu'avec des serveurs qui prennent en charge la fonction DECRYPT_BIN ou DECRYPT_CHAR.

z/OS FIELDPROC

La réplication Q prend en charge les colonnes définies dans des tables DB2 pour z/OS avec des procédures de zone (FIELDPROC) pour transformer les valeurs. Le sous-système DB2 contenant les tables avec des colonnes FIELDPROC doit avoir le correctif APAR PK75340 ou un niveau de correctif supérieur.

Si possible, pour améliorer les performances, créez l'index suivant dans votre table SYSIBM.SYSFIELDS :

```
CREATE INDEX "SYSIBM"."FIELDSX"  
ON "SYSIBM"."SYSFIELDS"  
(TBCREATOR ASC,  
TBNAME ASC,  
NAME ASC)  
USING STOGROUP SYSDEFLT PRIQTY 100 SECQTY 100  
CLOSE NO;  
COMMIT;
```

Restrictions sur les types de données

Actuellement, les données suivantes ne peuvent pas être répliquées ou publiées :

- Type de données spatiales
- **z/OS** Toute colonne pour laquelle une clause VALIDPROC est définie
- Vous pouvez dupliquer les types de données BINARY ou VARBINARY lorsque la source et la cible résident sous z/OS. La répllication de ces types de données entre une source z/OS et une cible DB2 pour Linux, UNIX et Windows ou une cible fédérée n'est pas prise en charge. Les types de données BINARY et VARBINARY sont pris en charge en tant que cibles d'expressions source uniquement si le type de données source est CHAR, VARCHAR, GRAPHIC, VARGRAPHIC ou ROWID.

Vous pouvez répliquer ou publier les types de données suivants sous certaines conditions :

LONG VARCHAR et LONG VARGRAPHIC

Les colonnes avec des types de données caractère de variable longue (LONG VARCHAR) et graphique de variable longue (LONG VARGRAPHIC) sont assorties des restrictions suivantes :

- LONG VARCHAR et LONG VARGRAPHIC ne peuvent être répliquées depuis DB2 pour Linux, UNIX, et Windows sur DB2 pour z/OS. Les zones dans DB2 for z/OS qui contiennent des caractères de variable longue présentent une longueur maximum plus petite que les zones sous DB2 for Linux, UNIX et Windows. Par conséquent, la répllication de ces types de zone dans DB2 for z/OS à partir de DB2 for Linux, UNIX et Windows peut provoquer une troncature.
- Lorsque vous spécifiez DATA CAPTURE CHANGES pour une table source lors de la création de la table, toute colonne LONG VARCHAR et LONG VARGRAPHIC est automatiquement activée pour la répllication. Si vous ajoutez des colonnes LONG VARCHAR à la table après avoir créé un abonnement Q et que la table n'avait plus de colonnes LONG, vous devez utiliser l'instruction ALTER TABLE pour activer DATA CAPTURE CHANGES INCLUDE LONGVAR COLUMNS pour les nouvelles colonnes LONG VARCHAR ou LONG VARGRAPHIC.

types de données définis par l'utilisateur

Vous pouvez répliquer ou publier des types de données distincts définis par l'utilisateur mais pas des types de données structurés et de référence et définis par l'utilisateur. Les types de données distincts définis par l'utilisateur (types de données distincts dans DB2) sont convertis en type de données de base avant d'être répliqués. Si la table cible est créée au moment où l'abonnement Q est créé, les types de données distincts définis par l'utilisateur sont convertis en type de données de base dans la nouvelle table cible.

type de données GRAPHIC

Les colonnes comportant le type de données GRAPHIC à la source et à la cible risquent de ne pas correspondre lorsque vous utilisez l'utilitaire asntdiff pour vérifier que les tables source et cible sont les mêmes. Les données GRAPHIC contenues dans les colonnes DB2 sont suivies de caractères de remplissage. Ces caractères peuvent prendre la forme d'espaces à un ou deux octets, selon la page de codes dans laquelle la base de données a été créée. Ces caractères de remplissage peuvent être à l'origine de la non correspondance des données des tables source et cible,

surtout si celles-ci sont associées à des pages de codes différentes. Ces caractères de remplissage s'appliquent uniquement aux types de données GRAPHIC et non aux autres types de données graphiques tels que VARGRAPHIC ou LONG VARGRAPHIC.

Pour comparer les colonnes contenant des données de type GRAPHIC, vous devez supprimer les caractères de remplissage dans les données avant de comparer les tables source et cible à l'aide de la fonction scalaire DB2 suivante :

```
rtrim(colonne)
```

Cette fonction élimine les différences de page de codes pour les espaces à un ou deux octets et veille à ce que l'utilitaire asntdiff compare les données GRAPHIC de manière cohérente.

TIMESTAMP WITH TIME ZONE

Sous certaines conditions, il est possible de répliquer le type de données TIMESTAMP WITH TIMEZONE apparu dans DB2 for z/OS Version 10. Le tableau 22 représente les mappages pris en charge

Tableau 22. Mappages de colonnes pris en charge pour le type de données TIMESTAMP WITH TIMEZONE

		Colonne cible	
		TIMESTAMP WITH TIMEZONE	TIMESTAMP WITHOUT TIMEZONE
Colonne source	TIMESTAMP WITH TIMEZONE	Autorisé si la longueur de la colonne source est inférieure ou égale à la longueur de la colonne cible	Non autorisé
	TIMESTAMP WITHOUT TIMEZONE	Non autorisé	Autorisé

Remarque : DB2 pour Linux, UNIX et Windows ne prend pas en charge TIMESTAMP WITH TIMEZONE.

Considérations concernant les types de données d'objet LOB pour la répllication Q et la publication d'événements

Une répllication prend en charge les types de données des objets BLOB (objet binaire de grande taille), objets CLOB et objets DBCLOB différemment selon le réglage du paramètre **lob_send_option** de Q Capture et la taille des données des objets LOB.

Les valeurs du paramètre sont les suivantes :

I (en ligne)

Les données LOB sont envoyées dans le message de transaction. Cette méthode peut permettre d'améliorer les performances. Vous pourriez avoir besoin d'ajuster la valeur de l'option **max_message_size** pour la mappe de files de répllication afin de vous assurer que la taille des messages est suffisante pour détenir des valeurs LOB.

S (séparé)

Les données LOB sont envoyées à la suite du message de transaction dans un ou plusieurs messages LOB. Le nombre de messages dépend de la valeur de l'option **max_message_size**.

Quel que soit le réglage du paramètre **lob_send_option**, en démarrant avec DB2 10.1 for Linux, UNIX et Windows, le programme Q Capture peut lire des données LOB directement à partir du journal de reprise de DB2 plutôt que d'avoir à se connecter à la base de données et procéder à une extraction dans la table source. Cette méthode peut permettre également améliorer les performances.

Pour les bases de données source au niveau de DB2 version 10.1 ou des versions ultérieures, les données LOB sont lues à partir du journal dans les cas suivants :

- Les données tiennent dans une ligne de la table source, ligne affectée par le paramètre **INLINE LENGTH** dans la définition d'une colonne LOB. Ce paramètre indique la taille maximale en octets d'une valeur LOB qui peut être stockée dans une ligne de table de base.
- La colonne LOB a été créée avec l'attribut **LOGGED**, qui indique que les changements apportés à la colonne sont consignés dans le journal, même si la taille des données excède la valeur du paramètre **INLINE LENGTH**.

Si une valeur LOB ne tient pas dans un message, la valeur **LOB_TOO_BIG_ACTION** de la table **IBMQREP_SENDQUEUES** détermine quelle action Q Capture doit entreprendre :

Q (par défaut)

Le programme Q Capture suit l'action d'erreur définie pour la file d'attente d'envoi.

E Le programme Q Capture envoie des valeurs LOB vides si les données ne tiennent pas dans un seul message de transaction. Si la valeur de remplacement est trop volumineuse pour un message, Q Capture suit l'action d'erreur pour la file d'attente.

Avec **lob_send_option=S**, le programme Q Capture envoie des valeurs LOB pour toutes les colonnes LOB qui font partie d'un abonnement Q ou d'une publication pour toutes les lignes d'une transaction. Ce comportement engendre davantage de messages WebSphere MQ si une valeur LOB est mise à jour à plusieurs reprises dans une transaction ou si l'option **CHANGED_COLS_ONLY** de l'abonnement Q ou de la publication est définie sur **N**. Pour les mises à jour de clés, les valeurs LOB de toutes les colonnes LOB faisant partie d'un abonnement Q ou d'une publication sont envoyées sans tenir compte de la définition de **CHANGED_COLS_ONLY**.

Recommandation : Si vous répliquez le type de données **DBCLOB**, définissez **lob_send_option=I**, notamment si vous attendez des valeurs **DBCLOB** supérieures à 16 Ko.

Cibles fédérées : vous pouvez répliquer les objets LOB dans les cibles Oracle mais pas dans les cibles non DB2.

Autres informations importantes à propos de la réplification LOB :

- Pour répliquer les données LOB, vous devez disposer d'un index unique dans votre table source.
- Un objet LOB entier est répliqué ou publié, même si uniquement une petite partie de l'objet LOB est modifiée.

- La réplication Q et la publication d'événements ne prennent pas en charge DB2 Extensions pour le texte, l'audio, la vidéo, l'image ou d'autres extensions où des fichiers de contrôle supplémentaires associés aux données de colonne d'objet LOB de l'extension sont gérés en dehors de la base de données.
- Les valeurs antérieures des colonnes LOB ou ROWID ne sont pas répliquées ou publiées. Lorsque le programme Q Capture voit l'indication d'un changement LOB (un descripteur LOB) dans le journal de reprise de DB2, le programme Q Capture envoie la valeur LOB à partir de la table source.

type de données XML

DB2 Version 9.1 et supérieure prend en charge le stockage natif des documents XML. Vous pouvez répliquer ou publier les documents XML.

Les documents XML sont répliqués au sein du message de transaction, de la même manière que les autres colonnes. Si les documents XML sont volumineux, vous devrez peut-être accroître la taille maximale des messages pour les messages WebSphere MQ. Si un document XML dépasse la taille maximale des messages pour la file d'attente d'envoi, le document XML n'est pas envoyé. Vous pouvez définir le paramètre XML_TOO_BIG_ACTION pour la file d'attente d'envoi dans la table IBMQREP_SENDQUEUES des façons suivantes pour définir l'action effectuée par Q Capture :

Q (par défaut)

Le programme Q Capture suit l'action d'erreur définie pour la file d'attente d'envoi.

E Le programme Q Capture envoie une marque de réservation XML. Si la marque de réservation XML est trop volumineuse pour un message, Q Capture suit l'action d'erreur pour la file d'attente.

Le programme Q Apply ajoute une ligne à la table IBMQREP_EXCEPTIONS à chaque fois que des documents XML ne peuvent pas être envoyés.

Conditions requises

Pour répliquer les données XML, vous devez disposer d'un index unique dans votre table source.

En complément de la réplication vers le type de données XML DB2, vous pouvez répliquer des documents XML vers des cibles fédérées qui prennent en charge les types de données XML ou vous pouvez mapper la colonne XML source vers une colonne CLOB ou BLOB au niveau de la cible.

Restrictions

Le type de données XML comporte les restrictions suivantes :

- Vous ne pouvez pas filtrer les lignes en fonction du contenu d'un document XML.
- Vous ne pouvez pas répliquer de colonnes XML que vous avez définies comme index unique.
- Les documents ne sont pas validés par le programme Q Apply.
- Vous ne pouvez pas répliquer les enregistrements de schéma XML.

- **z/OS** Sous z/OS, vous ne pouvez pas effectuer un chargement automatique pour les tables qui contiennent des colonnes XML. Le programme DSNUTILS qui effectue le chargement automatique ne prend pas en charge le type de données XML.
- Vous ne pouvez pas répliquer des colonnes XML à partir des sources Oracle.
- Les index à entrées uniques définis par l'utilisateur sur des expressions XPATH sur des colonnes XML ne sont pas pris en charge. Pour répliquer ces colonnes, supprimez l'index à entrées uniques et utilisez un index qui n'est pas à entrées uniques.

Réplication entre colonnes XML et LOB

La réplication Q prend en charge les mappages entre colonnes XML et objet LOB, avec parfois quelques restrictions.

Pas de restriction

Le mappage suivant est pris en charge sans restriction :

- BLOB sur XML

Pris en charge avec des restrictions

Les mappages suivants sont pris en charge, mais aucun chargement automatique (type I) n'est autorisé car le déplacement des données de colonnes XML dans des colonnes BLOB ou CLOB n'est pas pris en charge par les utilitaires LOAD ou IMPORT de DB2 qui sont utilisés dans les chargements automatiques :

- XML sur BLOB
- XML sur CLOB

Vous pouvez utiliser les chargements automatiques avec CLOB sur XML, mais si vous spécifiez les utilitaires EXPORT et LOAD les bases de données source et cible doivent avoir la même page de code :

- CLOB sur XML

Avertissement : Si vous répliquez un objet CLOB sur XML, la perte de données peut se produire si les deux conditions suivantes sont vraies :

- La base de données source utilise une page de code non-UTF-8 et la base de données cible utilise une page de code 1208 (codeset UTF-8).
- L'abonnement Q indique un chargement automatique qui utilise les utilitaires EXPORT et IMPORT.

Dans cette situation, vous devez utiliser l'utilitaire LOAD from CURSOR pour l'abonnement Q.

Non pris en charge

Les mappages suivants ne sont pas pris en charge :

- XML sur DBCLOB
- DBCLOB sur XML

Réplication de nouveaux types de données DB2 version 9.7 (Linux, UNIX, Windows)

La réplication Q prend en charge les nouveaux types de données suivants, introduits avec la version 9.7 de DB2 for Linux, UNIX pour Windows, pour faciliter la migration des applications vers DB2.

Certains des nouveaux types de données nécessitent des considérations spécifiques en matière d'environnement de réplication. Les sections suivantes fournissent des informations détaillées :

- «TIMESTAMP et précision étendue»
- «DATE et option de compatibilité», à la page 240
- «NUMBER», à la page 240

TIMESTAMP et précision étendue

La réplication Q prend en charge la réplication de données TIMESTAMP avec une précision étendue, allant de TIMESTAMP(0) à TIMESTAMP(12). Si les bases de données source et cible et les programmes Q Capture et Q Apply sont de version 9.7 ou une version ultérieure, les données source sont complétées ou tronquées sur la base de données cible pour satisfaire un mappage de colonnes TIMESTAMP non correspondantes. Le mappage de colonnes TIMESTAMP non correspondantes n'est pris en charge que pour une réplication unidirectionnelle.

Si un programme Q Capture V9.7 envoie des données à un programme Q Capture plus ancien (COMPATIBILITY inférieure à 0907 dans la table IBMQREP_CAPPARMS), Q Capture envoie uniquement les données TIMESTAMP(6) pour qu'elles correspondent à la capacité de la cible. Dans cette situation, Q Capture peut devoir compléter ou tronquer les données source, selon la taille de la colonne TIMESTAMP source V9.7.

Par exemple, si vous avez répliqué une source V9.7 vers une cible V9.5 et qu'une table source comportait une colonne TIMESTAMP(12), le programme Q Capture 9.7 tronque six chiffres de la portion concernant les fractions de secondes de la valeur TIMESTAMP. La troncature est nécessaire car DB2 V9.5 ne prend pas en charge la précision étendue, pour que la portion des valeurs TIMESTAMP des bases de données V9.5 relative aux fractions de secondes soit égale à TIMESTAMP(6). Le tableau 23 affiche une valeur pour la source et la valeur tronquée obtenue pour la cible.

Tableau 23. Troncature de TIMESTAMP(12) lors de la réplication

Valeur source dans TIMESTAMP(12)	Valeur cible dans TIMESTAMP(6)
2009-07-10-10.33.42.458499823012	2009-07-10-10.33.42.458499

Dans ce même scénario, si les données source sont dans une plage comprise entre TIMESTAMP(0) et TIMESTAMP(5), DB2 complète automatiquement les données au niveau antérieur à la version 9.7, avec six chiffres pour les fractions de secondes.

z/OS **Remarque** : lors de la gestion de ces nouveaux types de données, la réplication Q traite une source ou une cible DB2 pour z/OS comme s'il s'agissait de DB2 pour Linux, UNIX et Windows version 9.5 ou une version ultérieure.

Si le programme Q Apply est à la version 9.7 ou une version ultérieure et que le programme Q Capture est plus ancien, DB2 complète ou tronque automatiquement les valeurs TIMESTAMP pour qu'elles correspondent à la précision de la colonne TIMESTAMP cible. Le tableau 24, à la page 240 donne un exemple de remplissage.

Tableau 24. Remplissage d'une valeur *TIMESTAMP* plus ancienne lors de la réplication

Valeur source dans <i>TIMESTAMP</i>	Valeur cible dans <i>TIMESTAMP(12)</i>
2009-07-10-10.33.42.458499	2009-07-10-10.33.42.458499000000

DATE et option de compatibilité

L'option de compatibilité de date stocke le type *DATE* avec une portion d'heure supplémentaire (HH:MM:SS). Ce format est conforme à la représentation de date des autres systèmes de gestion de bases de données relationnelles, comme Oracle, dont le type de données *DATE* inclut YYYY-MM-DD HH:MM:SS.

La réplication Q traite les bases de données sans compatibilité de date comme des bases de données DB2 antérieures à la version 9.7 et comme les sous-systèmes DB2 pour z/OS. Lorsque la compatibilité de date est activée, DB2 gère les colonnes de type *DATE* de la même façon qu'il gère les colonnes de type *TIMESTAMP(0)*.

Activez la prise en charge de *DATE* comme *TIMESTAMP(0)* en définissant la position binaire 7 (0x40) de la variable de registre *DB2_COMPATIBILITY_VECTOR* avant de créer une base de données. Avec la réplication Q unidirectionnelle, vous pouvez créer les mappages de colonnes suivants entre *DATE* et *TIMESTAMP(0)* :

DATE vers *TIMESTAMP(0)*

Si la compatibilité de date n'est pas activée sur la base de données source, la valeur cible est complétée en YYYY-MM-DD-00:00:00.

***TIMESTAMP(0)* vers DATE**

Si la compatibilité de date n'est pas activée sur la base de données cible, la valeur *TIMESTAMP(0)* est tronquée en YYYY-MM-DD.

NUMBER

Le type de données *NUMBER* prend en charge les applications utilisant le type de données *NUMBER* Oracle. DB2 traite les données *NUMBER* de façon interne comme *DECFLOAT* si aucune précision ou échelle n'est indiquée et comme *DECIMAL* avec une précision ou une échelle si ces attributs sont indiqués.

Comme la réplication Q prend déjà en charge *DECFLOAT* et *DECIMAL*, pour la réplication unidirectionnelle, vous pouvez mapper des colonnes définies avec l'un de ces types numériques avec l'autre : *NUMBER* vers *DECFLOAT* ou *DECIMAL*, *DECFLOAT* vers *NUMBER* ou *DECIMAL* et *DECIMAL* vers *NUMBER* ou *DECFLOAT*.

Réplication de tables avec des colonnes d'identité

La réplication Q autorise les colonnes d'identité dans les tables source et cible mais, en raison de restrictions DB2, des étapes supplémentaires peuvent être nécessaires si votre table source comporte des colonnes définies avec la clause *AS IDENTITY GENERATED ALWAYS*.

Les colonnes d'identité sont gérées différemment par la réplication, selon si elles se trouvent dans la table source ou la table cible :

Table source

Si vous disposez d'une colonne d'identité dans une table source et que vous voulez la répliquer dans une table cible, créez un abonnement Q comme d'habitude. La table cible est créée avec des colonnes numériques

pour contenir les valeurs. Par exemple, une colonne source définie en tant que `GENERATE ALWAYS` peut être répliquée vers une colonne `BIGINT` dans la table cible. Les colonnes de la table cible ne peuvent pas être des colonnes d'identité et vous ne pouvez donc pas répliquer une colonne d'identité d'une table source dans une colonne d'identité d'une table source.

Table cible

Si vous disposez d'une colonne d'identité dans une table cible, n'incluez pas cette colonne dans votre abonnement Q. La colonne est automatiquement remplie lorsque la réplication insère la table cible ou la met à jour. Le comportement de la colonne d'identité est indiqué pour les insertions et les mises à jours par tout autre application. Si vous répliquez la même table source vers plusieurs tables cible ayant des colonnes d'identité, les valeurs d'identité de ces tables cible sont indépendantes les unes des autres.

DB2 ne permet pas l'insertion dans des colonnes définies avec la clause `AS IDENTITY GENERATED ALWAYS` car cette clause n'est pas prise en charge pour les tables cible de réplication Q. Il existe cependant des options pour la réplication de ces colonnes :

- Créer la table cible sans la clause `IDENTITY`.
- Créer la table cible avec une colonne définie comme `AS IDENTITY GENERATED BY DEFAULT`.

Pour les colonnes définies avec `AS IDENTITY GENERATED BY DEFAULT`, la plage de valeurs doit être différentes entre la source et la cible car DB2 ne garantit pas l'unicité des colonnes d'identité entre deux bases de données DB2 distinctes.

Par exemple, la colonne d'identité d'un site peut être définie sur des nombres pairs (`START WITH 2, INCREMENT BY 2`) et sur l'autre site, la colonne d'identité peut être définie sur des nombres impairs (`START WITH 1, INCREMENT BY 2`). Vous pouvez également affecter une plage à un site (par exemple, 1 à 10000 sur un site et 20000 à 40000 sur l'autre). L'approche pair-impair permet de s'assurer qu'en cas de conflit, deux lignes différentes ayant par accident la même clé d'identité générée ne s'écrase pas mutuellement lorsque l'action en cas de conflit est de forcer l'application de la modification.

Le type de données de la colonne d'identité (`SMALLINT`, `INTEGER` ou `BIGINT`) doit être déterminé en fonction des besoins de l'application (nombre le plus élevé attendu dans la colonne par exemple).

Les colonnes d'identité doivent être de type `NO CYCLE` si les nombres ne peuvent pas être réutilisés. Mettez un plan en place sur les actions à effectuer lorsque la valeur maximale est atteinte (`SQLSTATE 23522`). Si vous utilisez `CYCLE`, assurez-vous qu'une nouvelle utilisation d'un numéro ne génère pas d'incident pour une utilisation existante de ce numéro, y compris lors de la réplication.

Chapitre 14. Utilisation des commandes et des scripts générés par les outils d'administration de la réplication

Le programme de ligne de commande ASNCLP et le centre de réplication génèrent des scripts SQL pour définir et changer les objets de réplication. Le Centre de réplication génère des commandes opérationnelles pour les tâches telles que le démarrage et l'arrêt des programmes de réplication, l'élagage des tables de contrôle, la modification des paramètres ou la vérification du statut du programme.

Vous pouvez utiliser les outils d'administration de la réplication pour exécuter les scripts et les commandes qu'ils génèrent ou vous pouvez enregistrer les scripts et les commandes, les modifier et les exécuter plus tard.

Exécution et enregistrement des scripts générés par les outils d'administration de la réplication

Pour créer une réplication et publier les objets, vous exécutez des scripts SQL générés par le programme de ligne de commande ASNCLP ou le Centre de réplication. Vous pouvez modifier les scripts, utiliser les outils pour exécuter les scripts ou exécuter les scripts à partir d'une ligne de commande DB2.

A propos de cette tâche

Lors de la modification des scripts SQL générés, veillez à ne pas changer les caractères de fin. Veillez également à ne pas changer les séparateurs de scripts si plusieurs scripts sont enregistrés dans un même fichier.

Vous pouvez adapter les scripts SQL à votre environnement pour effectuer les tâches suivantes :

- Créer des copies multiples de la même action de réplication, personnalisées pour plusieurs serveurs.
- Regrouper des définitions et les exécuter en tant que travail par lots.
- Différer la réplication à une heure spécifiée.
- Créer des bibliothèques de scripts SQL pour la sauvegarde, pour la personnalisation en fonction du site ou pour l'exécution autonome sur des sites distribués, tel qu'un environnement parfois connecté.

Procédure

1. Utilisez l'une des méthodes suivantes pour exécuter ou enregistrer les scripts générés par les outils d'administration de la réplication:

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande SET RUN SCRIPT pour décider d'exécuter automatiquement les instructions SQL générées par chacune des commandes de tâche ASNCLP avant de traiter la commande suivante ou de les exécuter manuellement plus tard via une invite de commande DB2.</p> <p>NOW Cette option exécute automatiquement le SQL généré.</p> <p>LATER Cette option enregistre le SQL généré dans un fichier. Vous pouvez préciser le chemin du fichier et son nom avec la commande SET OUTPUT.</p> <p>Par exemple, la commande suivante prévoit l'exécution automatique du script SQL et l'arrêt du traitement des commandes ASNCLP si une erreur survient :</p> <pre>SET RUN SCRIPT NOW STOP ON SQL ERROR ON</pre>
Centre de réplication	<p>Utilisez la fenêtre Exécution immédiate et enregistrement du script.</p> <p>Le bouton radio Exécution immédiate est initialement sélectionné de sorte que lorsque vous cliquez sur OK, le Centre de réplication émet des instructions SQL pour créer ou modifier un objet.</p> <p>Ensuite, la fenêtre sélectionne la dernière option de traitement que vous avez choisie. Par exemple, si vous décidez d'enregistrer les scripts dans un fichier, lors du prochain affichage de cette fenêtre, le bouton radio Enregistrer dans le fichier est sélectionné et les mêmes système et chemin d'accès apparaissent dans la zone Spécifications de l'exécution.</p> <p>Linux UNIX Windows Vous pouvez également planifier les scripts SQL afin de les exécuter en tant que tâches dans le Centre de gestion des tâches sous Linux, UNIX et Windows.</p> <p>Le bouton Appliquer vous permet d'exécuter le script sans fermer la fenêtre, si bien que vous pouvez toujours enregistrer le script en tant que fichier ou tâche.</p>

2. Facultatif : Exécutez les fichiers contenant les scripts SQL via la ligne de commande DB2 en utilisant l'une des méthodes suivantes :
- Utilisez cette commande si le caractère de fin du script est un point-virgule (;):

```
db2 -tvf nom_fichier
```
 - Utilisez cette commande si le script SQL utilise un autre caractère en tant que délimiteur :

```
db2 -tdcaract -vf nom de fichier
```

où *caract* représente un caractère de fin tel que #.

Si vous exécutez les scripts SQL à partir d'une ligne de commande DB2, vous devez vous connecter aux serveurs manuellement lors de l'exécution du script SQL. Le script est généré avec des déclarations CONNECT. Avant d'exécuter le script SQL, vous devez modifier les instructions SQL pour préciser l'ID utilisateur et le mot de passe pour le serveur. Par exemple, recherchez une ligne qui ressemble à l'exemple suivant et ajoutez vos informations en écrasant les paramètres fictifs (XXXX) :

```
CONNECT TO nom_basededonnées USER XXXX USING XXXX ;
```

Exécuter et enregistrer les commandes (Centre de réplication)

Il est possible d'exécuter des commandes directement à partir du Centre de réplication, d'enregistrer et d'exécuter les commandes en tant que fichiers de traitement par lots via la ligne de commande ou d'enregistrer les commandes en tant que tâches pour le Centre de gestion des tâches.

Procédure

Pour gérer les commandes générées par le Centre de réplication, utilisez la fenêtre Exécution immédiate ou Enregistrer la commande.

1. Facultatif : Modifiez la commande dans la zone de texte.
2. Choisissez l'une des options suivantes :
 - Pour exécuter la commande immédiatement, cliquez sur **Exécution immédiate**, renseignez les zones **Spécifications de l'exécution** puis cliquez sur **OK**.
 - Pour enregistrer la commande, cliquez sur **Enregistrer dans le fichier**, saisissez les champs **Spécifications de l'exécution** puis, cliquez sur **OK**. Le centre de réplication crée un fichier avec le nom et l'extension que vous définissez ou vous pouvez enregistrer la commande dans un fichier existant.
 - Pour enregistrer la commande en tant qu'objet de tâche pour le Centre de gestion des tâches, cliquez sur **Enregistrer en tant que tâche**, renseignez les zones **Spécifications de l'exécution** puis cliquez sur **OK**.

Chapitre 15. Utilisation d'un programme Q Capture

Un programme Q Capture enregistre des transactions ou des modifications au niveau des lignes des tables source faisant partie d'un abonnement Q ou d'une publication, puis envoie ces données transactionnelles sous forme de messages aux files d'attente WebSphere MQ.

Vous pouvez exploiter un programme Q Capture à l'aide du Centre de réplication, des commandes et des services système, et modifier les paramètres d'exploitation de Q Capture de plusieurs manières.

Démarrage d'un programme Q Capture

Vous démarrez un programme Q Capture pour commencer à capturer les transactions ou les modifications au niveau des lignes à partir du journal de récupération DB2 pour les nouveaux abonnements Q ou les abonnements Q actifs ou les publications, et à envoyer les données transactionnelles comme messages sur les files d'attente WebSphere MQ.

Avant de commencer

- Si vous démarrez un programme Q Capture à partir d'un poste de travail à distance, configurez les connexions sur le serveur Q Capture.
- Créez un gestionnaire de files d'attente WebSphere MQ, des files d'attente et d'autres objets requis.
- Assurez-vous que vous avez les droits requis pour les objets de réplication Q et les objets de publication d'événements ainsi que pour les objets WebSphere MQ.
- Créez des tables de contrôle pour le schéma Q Capture approprié.
- Configurez la base de données ou le sous-système source pour travailler avec le programme Q Capture.

Important : Assurez-vous que la consignation des archives est activée sur la base de données que vous utilisez comme serveur Q Capture. Servez-vous de la fenêtre Activer la consignation des archives du Centre de réplication pour configurer le serveur Q Capture pour la consignation des archives et effectuez une sauvegarde hors ligne de la base de données.

- Si le moindre abonnement Q spécifiant le chargement automatique et utilisant l'utilitaire EXPORT est à l'état N (nouveau) ou A (actif), créez un fichier de mots de passe sur le serveur Q Apply pour permettre à l'utilitaire de se connecter au serveur Q Capture.

A propos de cette tâche

Lors du démarrage initial d'un programme Q Capture sans spécification du mode de démarrage, le programme utilise le mode de démarrage par défaut, warmsi. Par ce mode, le programme essaie de lire le journal à l'emplacement où il s'était arrêté. Le programme démarrant pour la première fois, Q Capture passe en mode de démarrage à froid et démarre le traitement des abonnements Q ou des publications qui sont à l'état N (nouveau) ou A (actif). Tous les abonnements Q ou publications qui sont à l'état I (inactif) doivent être activés pour que le programme commence à capturer les modifications.

Vous pouvez démarrer un programme Q Capture même si aucun abonnement Q ou publication n'est à l'état A (actif). Lorsque vous activez les abonnements Q ou les publications, le programme Q Capture commence à capturer les modifications.

Lorsque vous démarrez un programme Q Capture, vous pouvez définir les valeurs des paramètres d'initialisation et le programme utilisera les nouvelles valeurs jusqu'à l'une des actions suivantes :

- Changer les valeurs des paramètres pendant que le programme est exécuté.
- Arrêter et redémarrer le programme, ce qui l'invite à lire la table IBMQREP_CAPPARMS et utiliser les valeurs sauvegardées ici.

Procédure

Pour démarrer un programme Q Capture, appliquez l'une des méthodes suivantes :

Méthode	Description
<p>z/OS</p> <p>Console ou TSO z/OS</p>	<p>Sous z/OS, vous pouvez démarrer un programme Q Capture avec JCL ou comme tâche lancée par le système. Vous pouvez définir de nouvelles valeurs de paramètre d'appel lorsque vous démarrez un programme Q Capture à l'aide du langage JCL.</p> <p>La longueur totale des paramètres z/OS, que vous pouvez indiquer dans la zone PARM=, est limitée à 100 octets. Pour dépasser cette limite, les programmes de réplication permettent désormais de préciser autant de paramètres que nécessaire dans le fichier SYSIN.</p> <p>Lorsque l'appel JCL comprend l'instruction de définition de données SYSIN, les éléments spécifiés dans le fichier SYSIN sont automatiquement concaténés par le programme Q Capture aux paramètres PARM=. Vous pouvez indiquer uniquement les paramètres Q Capture dans le fichier SYSIN. Les paramètres LE doivent être indiqués dans la zone PARM= ou LE _CEE_ENVFILE=DD, suivis d'une barre oblique (/).</p> <p>Exemple :</p> <pre> /* asterisk indicates a comment line // QCAP EXEC PGM=ASNQCAP,PARMS='LE/Q Capture parameters' /* Parameters can be any or no LE parameters and any or /* no Q Capture parameters //SYSIN DD * /* additional Q Capture parameters, one or more /* parameters on each line CAPTURE_SERVER=DSN!! CAPTURE_SCHEMA=CAPCAT DEBUG=Y LOGSTDOUT=N </pre>
<p>Commande asnqcap</p>	<p>Utilisez la commande asnqcap pour démarrer un programme Q Capture et définissez les paramètres d'initialisation. Par exemple :</p> <pre> asnqcap capture_server=<i>nom_serveur</i> capture_schema=<i>schéma</i> <i>paramètres</i> </pre> <p>Où <i>nom_serveur</i> est le nom de la base de données ou du sous-système contenant la table de contrôle Q Capture, <i>schéma</i> identifie le programme Q Capture que vous souhaitez démarrer et <i>paramètres</i> correspond à un ou plusieurs des paramètres que vous pouvez définir à l'initialisation.</p>

Méthode	Description
Windows Services Windows	Vous pouvez créer un service de réplication sous les systèmes d'exploitation Windows pour démarrer le programme Q Capture automatiquement quand le système est lancé.

Vous pouvez vérifier si un programme Q Capture a démarré en appliquant une des méthodes suivantes :

- Recherchez dans le fichier journal de diagnostic Q Capture (*serveur_capture.schéma_capture.QCAP.log* sur z/OS et *instancedb2.serveur_capture.schéma_capture.QCAP.log* sur Linux, UNIX et Windows) un message indiquant que le programme capture les modifications.
- Recherchez dans la table IBMQREP_CAPTRACE un message indiquant que le programme capture les modifications.
- **z/OS** Si vous êtes en mode de traitement par lots, examinez la console z/OS ou le journal des travaux z/OS pour voir si des messages indiquent que le programme a démarré.
- Utilisez la fenêtre Messages Q Capture dans le Centre de réplication pour afficher un message indiquant que le programme a démarré. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture contenant le programme Q Capture dont vous souhaitez afficher les messages et sélectionnez **Rapports > Messages Q Capture**.

Démarrage de Q Capture à partir d'un emplacement connu dans le journal DB2

Vous pouvez utiliser des paramètres de ligne de commande pour démarrer le programme Q Capture à un emplacement connu dans le journal DB2 sans déclencher un chargement de la table cible.

A propos de cette tâche

D'ordinaire, lorsque le programme Q Capture est arrêté, vous utilisez un redémarrage à chaud pour commencer à lire le journal de récupération DB2 où le programme Q Capture s'est arrêté. Vous utilisez généralement un redémarrage à froid pour commencer à lire à la fin du journal. Un redémarrage à froid invite automatiquement le programme Q Apply à charger la table cible avec les dernières données depuis la source.

Dans certaines situations, vous allez peut-être redémarrer le programme Q Capture à partir d'un emplacement connu dans le journal. Par exemple, dans un scénario de reprise en ligne à haute disponibilité, vous pouvez enregistrer les informations à partir du journal DB2 sur un serveur, puis, utiliser une commande pour commencer à capturer les données sur le serveur de secours une fois que le second serveur a pris la suite.

Procédure

Pour démarrer un programme Q Capture à partir d'un emplacement connu dans le journal DB2 :

1. Déterminez l'emplacement du journal où vous souhaitez que Q Capture démarre la lecture. Vous aurez besoin des valeurs pour deux paramètres de commande :

lsn Le numéro d'ordre du journal de la plus ancienne transaction non validée que vous souhaitez capturer.

maxcmtseq

Le numéro d'ordre du journal de la transaction validée la plus récente placée dans la file d'attente.

2. Utilisez JCL ou la commande **asnqcap** et spécifiez les paramètres **lsn** et **maxcmtseq**.

Vous devez utiliser les deux paramètres dans la même instruction ou appel de commande JCL, et vous ne pouvez pas utiliser ces paramètres si la valeur de **startmode** est froid.

Par exemple, pour démarrer le programme Q Capture sur un serveur nommé SAMPLE avec une valeur **lsn** de 0000:C60F:2722:E137:0001 et une valeur **maxcmtseq** de 0000:C60F:2722:E080:0001, utilisez la commande suivante :

JCL

```
//QCAPDEC EXEC PGM=ASNQCAP,REGION=0M,TIME=NOLIMIT,
// PARM='STORAGE(FF,FF,FF)/SSTR CAPTURE_SCHEMA=QDECODER
//          STALE=20 STARTALLQ=N'
.
.
.
//SYSIN DD *
          MAXCMTSEQ=0000:C60F:2722:E080:0001
          LSN=0000:C60F:2722:E137:0001
//SYSTEM DD DUMMY
```

commande asnqcap

```
asnqcap SAMPLE LSN=0000:0000:0000:115b:7704
MAXCMTSEQ=41c2:2264:0000:0004:0000
```

Les exemples suivants montrent la manière dont vous pouvez redémarrer Q Capture à partir des différents emplacements dans le journal et trouver les valeurs de **lsn** et **maxcmtseq** dans différents scénarios.

Exemple 1 :

Pour démarrer à partir de la fin du journal sans déclencher une charge (régénération intégrale) de la table cible, indiquez l'une des valeurs suivantes dans la commande **asnqcap**, selon votre version de DB2 :

Version 9.7 et précédentes

lsn=FFFF:FFFF:FFFF:FFFF:FFFF et **maxcmtseq=FFFF:FFFF:FFFF:FFFF:FFFF**.

Version 10.1 ou supérieure avec le paramètre **compatibility de 1001 ou supérieur, ou version 9.8**

lsn=FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF et **maxcmtseq=FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF**.

Exemple 2 :

Dans certains cas, vous risquez de ne pas pouvoir utiliser un redémarrage à chaud, par exemple, si le message de redémarrage ou la file d'attente de redémarrage est perdue ou endommagée. Pour trouver les valeurs de **lsn** et **maxcmtseq** à l'endroit où le programme Q Capture s'est arrêté de lire le journal, appliquez l'une des méthodes suivantes :

Méthode	Description
Commande asnqfmt	<p>Exécutez la commande asnqfmt pour mettre en forme le message de redémarrage qui contient l'emplacement où Q Capture s'est arrêté de lire.</p> <p>Définissez le gestionnaire de files d'attente avec lequel le programme Q Capture travaille et le nom de la file d'attente de redémarrage. Par exemple :</p> <pre>asnqfmt asn.qml.restartq qml</pre> <p>Dans le résultat de la commande, cherchez les valeurs dans les zones restartLSN et qRestartMsg.lastCommitSEQ. Par exemple :</p> <pre>restartLSN=0000:0000:0000:03ab:65ab qRestartMsg.lastCommitSEQ: 4a02:3396:0000:0002:0000</pre> <p>Vous utilisez la valeur de qRestartMsg.restartLSN pour le paramètre lsn et la valeur de qRestartMsg.lastCommitSEQ pour le paramètre maxcmtseq.</p> <p>Remarque : Depuis la version 9.5, groupe de correctifs 3, le format du message de redémarrage a été modifié pour indiquer les points de redémarrage pour chaque file d'attente d'envoi utilisée par le programme Q Capture. Si vous utilisez une version de ce niveau ou d'un niveau supérieur, le LSN de redémarrage et l'indicateur de dernière validation apparaissent comme illustré dans l'exemple suivant :</p> <pre>sendq lastCommitSEQ: 4a02:3396:0000:0002:0000 sendq restartLSN=0000:0000:0000:03ab:65ab</pre> <p>Vous utilisez la valeur de sendq restartLSN pour le paramètre lsn et la valeur de sendq lastCommitSEQ pour le paramètre maxcmtseq.</p>
Journal de diagnostic Q Capture	<p>Recherchez le message ASN7109I dans le journal. Le fichier journal est trouvé dans le répertoire défini par le paramètre capture_path ou par défaut dans SQLLIB\bin.</p> <p>La valeur de lsn est décrite comme étant "le plus petit numéro d'ordre du journal d'une transaction à valider." La valeur de maxcmtseq est décrite comme étant "le numéro d'ordre du journal le plus élevé d'une transaction validée avec succès."</p> <p>Par exemple :</p> <pre>2006-01-11-16.12.49.583183 <asnqwk>ASN7109I "Q Capture" : "ASN" : "WorkerThread" : Lors de l'arrêt du programme, le numéro d'ordre du journal le plus élevé d'une transaction réussie est "43C5:9F00:0000:0001:0000" et le plus petit numéro d'ordre d'une transaction à valider est "0000:0000:0000:2B39:606D".</pre>

Exemple 3 :

Pour démarrer Q Capture à partir du moment où il a commencé sa dernière exécution, cherchez le message ASN7108I dans le journal de diagnostic Q Capture pour les valeurs de **lsn** et **maxcmtseq** au moment où Q Capture a démarré.

Par exemple :

```
2006-01-11-16.12.31.360146 <asnqwk> ASN7108I "Q Capture" : "ASN" : "WorkerThread" :
Lors de l'initialisation du programme, le numéro d'ordre du journal le plus
élevé d'une transaction validée avec succès est
"43C5:9EE5:0000:0001:0000" et le plus petit numéro d'ordre d'une
transaction à valider est "0000:0000:0000:2B38:82F6".
```

Vous pouvez également rechercher un message ASN7699I dans le journal de diagnostic Q Apply afin d'obtenir la valeur de **maxcmtseq** pour éviter que le programme Q Capture renvoie des transactions que Q Apply a déjà reçues.

Indiquer à Q Capture des points de redémarrage pour des files d'attente d'envoi individuelles ou des partitions individuelles de données (z/OS)

Lorsque vous démarrez un programme Q Capture en mode à chaud, vous pouvez redéfinir les informations de redémarrage global dans le message de redémarrage et spécifier des points de redémarrage différents pour les files d'attente d'envoi individuelles et les partitions individuelles dans une base de données source partitionnée.

Avant de commencer

L'ID utilisateur qui démarre le programme Q Capture doit avoir l'autorisation d'ouvrir et de lire le fichier de redémarrage.

A propos de cette tâche

Pour définir des points de redémarrage de files d'attente d'envoi individuelles ou de partitions de données individuelles, vous devez utiliser un paramètre d'initialisation de Q Capture permettant de remplacer les informations de redémarrage du message de redémarrage par les informations d'un fichier.

Le paramètre **override_restartq** ordonne à Q Capture de consulter le fichier de redémarrage et le message de redémarrage afin d'y rechercher le point à partir duquel il doit commencer la lecture du journal pour une file d'attente ou une partition de données en particulier lors de démarrage. Toute information de redémarrage figurant dans le fichier remplace l'information correspondante dans le message de redémarrage.

La possibilité de démarrer Q Capture à partir de points différents pour des files d'attente d'envoi individuelles ou des partitions individuelles de données est utile dans les situations suivantes :

- Vous souhaitez redémarrer Q Capture afin qu'il publie à nouveau des données pour une ou plusieurs cibles servies par des files d'attente d'envoi différentes sans affecter d'autres cibles. En cas de reprise après incident pendant laquelle le site principal se connecte à plusieurs sites cibles, la procédure de rémigration consistera à redémarrer Q Capture à partir de points différents pour chaque file d'attente d'envoi qui se connecte à un site cible différent, et ce, afin de renvoyer les modifications qui ont été perdues.
- Dans un environnement de bases de données partitionné, vous souhaitez démarrer Q Capture à partir d'un point connu du log différent du point que Q Capture a enregistré dans son message de redémarrage, ou bien vous souhaitez démarrer Q Capture en mode à chaud lorsque son message de redémarrage est perdu ou endommagé.
- Si le système de fichiers qui héberge des objets WebSphere MQ est perdu ou si le message de redémarrage Q Capture est perdu accidentellement et que plusieurs files d'attente d'envoi sont utilisées, vous pouvez redémarrer Q Capture à partir de points de redémarrage individuels.

Procédure

Pour démarrer Q Capture à partir d'un point spécifié du journal pour une ou plusieurs files d'attente d'envoi :

1. Effectuez l'une des actions suivantes :

- Si vous spécifiez pour la première un point de redémarrage au niveau de la file d'attente d'envoi ou de la partition, créez un fichier de redémarrage dans le répertoire spécifié par le paramètre **capture_path** ou dans le répertoire à partir duquel Q Capture a démarré si rien n'est indiqué pour **capture_path**. Utilisez la convention de nom suivante :

serveur_capture.schéma_capture.QCAP.QRESTART. Q Capture crée l'ensemble de données avec ces attributs :
DCB=(RECFM=FB,LRECL=128).

Linux UNIX Windows

instance_db2.serveur_capture.schéma_capture.QCAP.QRESTART

- Si le fichier QCAP.QRESTART existe déjà, mettez à jour son contenu comme suit.
2. Dans la sortie standard ou le fichier journal de Q Capture, recherchez le message d'information ASN7207I que Q Capture émet juste avant de s'arrêter. Ce message contient les informations suivantes pour chaque file d'attente d'envoi :
 - Nom de la file d'attente
 - Restart LSN (LSN le plus faible d'une transaction restant à valide)
 - commitSEQ (dernière transaction validée avec succès)
 - Numéro de partition pour les bases de données partitionnées sous Linux, UNIX et Windows

Voici un exemple de message ASN7207I :

ASN7207I "Q Capture" : "QALLTYPE" : "WorkerThread" :
Le contenu du fichier de remplacement restartq est le suivant :
queue name restart lsn maxcmtseq partition (uniquement pour les bases de données partitionnées).
QALLTYPEL12 0000:0000:4F07:634E:0000 0000:0000:4F00:547A:0000

3. Pour chaque file d'attente d'envoi ou partition de données pour lesquelles Q Capture doit capturer des transactions à partir du journal de reprise à un point spécifique, saisissez une ligne dans le fichier QCAP.QRESTART au format suivant :

Nom de la file d'attente d'envoi	Espace	restartLSN	Espace	commitSEQ	Espace (facultatif)	partition (facultative)

L'exemple suivant affiche le contenu d'un fichier de redémarrage pour deux files d'attente d'envoi, SQ1 et SQ2, sur les partitions de données 0 et 1 :

SQ1 0000:0000:4F07:634E:0000 0000:0000:4F00:547A:0000 0
SQ2 0000:0000:4FB9:0864:0000 0000:0000:4FB9:0864:0000 1

Les informations de partition de données ne sont pas requises pour z/OS, ni pour les bases de données non partitionnées sous Linux, UNIX et Windows. Le signe deux-points (:) est facultatif dans les valeurs restartLSN et commitSEQ.

4. Enregistrez le fichier QCAP.QRESTART.
5. Démarrez le programme Q Capture avec les paramètres **startmode=warmns** et **override_restartq=y**.

Q Capture lit le fichier de redémarrage lors de son initialisation et commence à lire le journal de récupération au point restartLSN spécifié pour chaque file d'attente à laquelle correspond des informations dans le fichier. Pour toutes les autres files d'attente d'envoi, Q Capture lit les informations globales de redémarrage dans le message de redémarrage.

En démarrant, Q Capture efface le fichier de redémarrage. En s'arrêtant, Q Capture écrit les informations de redémarrage les plus récentes dans le fichier et dans la sortie standard, son fichier journal et la table IBMQREP_CAPTRACE.

Remarque : Le paramètre **override_restartq=y** n'est pas compatible avec les paramètres **lsn** et **maxcmtseq**. Ces paramètres permettent de démarrer Q Capture à partir d'un point spécifié du journal pour des files d'attente d'envoi individuelles.

La prochaine fois que vous devrez démarrer Q Capture à partir d'un point spécifié pour une file d'attente d'envoi ou une partition de données en particulier, vous pouvez ouvrir le fichier QCAP.QRESTART et modifier les lignes correspondant aux files d'attente d'envoi à démarrer à partir d'un point en particulier. Pour cela, vous modifierez les valeurs de **restartLSN** ou de **commitSEQ**.

Lorsque **override_restartq** est spécifié au cours d'un démarrage à chaud, Q Capture suit le protocole suivant selon qu'un message de redémarrage figure ou non dans la file d'attente de redémarrage :

Le message de redémarrage existe

Le fichier de redémarrage doit contenir une ligne par file d'attente d'envoi dans le message de redémarrage dont vous souhaitez modifier **restartLSN** et **commitSEQ**. Q Capture remplace les informations de **restartLSN** et de **commitSEQ** de la file d'attente d'envoi dans le message de redémarrage par les informations de redémarrage correspondantes dans le fichier.

Le message de redémarrage n'existe pas

Le fichier de redémarrage doit contenir une ligne par file d'attente d'envoi figurant dans le message de redémarrage. Q Capture génère le message de redémarrage à partir des informations de redémarrage dans le fichier de redémarrage.

Considérations relatives à l'utilisation du mode démarrage à froid

Vous pouvez empêcher le chargement indésirable de la table cible et d'autres problèmes éventuels en tenant compte des aspects suivants concernant le mode démarrage à froid.

Lors du démarrage à froid d'un programme Q Capture à tout moment après le démarrage initial, le programme initie la lecture du journal DB2 à partir de la fin et non à partir du dernier point de redémarrage. Les résultats suivants peuvent se produire :

La source et la cible peuvent se désynchroniser, vous obligeant alors à charger la cible

Le programme Q Capture est susceptible d'ignorer des données d'enregistrements de journaux qu'il aurait normalement passés au programme Q Apply. Si ces enregistrements contiennent des mises à jour ou des insertions dans la table source, la seule manière de synchroniser les tables source et cible est de charger la table cible (parfois appelée régénération intégrale). Le chargement de la cible peut entraîner la perte des données d'historique.

Le chargement peut prendre un certain temps

Le chargement requiert une durée et un travail considérables pour les environnements contenant de nombreuses tables ou des volumes importants de données. Pour ces environnements, la régénération intégrale peut provoquer des indisponibilités coûteuses, notamment dans les

systèmes de production. Utilisez l'option de démarrage à froid en dernier recours, sauf si vous démarrez un programme Q Capture pour la première fois.

Vous pouvez observer les recommandations suivantes pour prévenir les démarrages à froid inattendus :

Définissez le paramètre startmode

Le paramètre **startmode** ne doit pas comporter la valeur cold (à froid). Spécifiez le mode de démarrage warmns ou warmsi chaque fois que possible. Lorsque vous spécifiez ces modes de démarrage, le programme Q Capture ne démarre pas à froid si les informations de démarrage à chaud sont indisponibles.

Surveillez le statut des programmes Q Capture à l'aide du moniteur d'alertes de réplication.

Par exemple, vous pouvez utiliser le critère d'alerte QCAPTURE_STATUS afin qu'il vous envoie une alerte par courrier électronique chaque fois que le moniteur détecte l'arrêt d'un programme Q Capture. Si le programme est indisponible pendant une longue période, vous devrez peut-être effectuer un démarrage à froid pour synchroniser les tables source et cible car il est possible que le programme Q Capture ne lise pas les enregistrements de journaux dans les temps alloués.

Conservez un nombre suffisant de données de journal DB2 et vérifiez que les données sont accessibles à Q Capture

Si les fichiers journaux ne sont pas accessibles au programme Q Capture, ce dernier ne peut pas poursuivre l'enregistrement des changements apportés aux tables source et un démarrage à froid peut s'avérer nécessaire.

Changement des paramètres de Q Capture

Vous pouvez modifier les paramètres d'exploitation de Q Capture lors du démarrage du programme, pendant son exécution ou en mettant la table de contrôle IBMQREP_CAPPARMS à jour.

Méthodes de changement des paramètres d'exploitation de Q Capture

Cette rubrique contient une brève description des trois différents modes de changement des paramètres, suivie d'un exemple vous aidant à mieux comprendre les différences entre chacun d'eux.

Changement des paramètres enregistrés dans la table IBMQREP_CAPPARMS

Les valeurs de paramètre du programme Q Capture sont enregistrées dans la table de contrôle IBMQREP_CAPPARMS. Une fois l'installation terminée, cette table est renseignée avec les valeurs par défaut du fabricant pour le programme. Un programme Q Capture lit la table lors de son démarrage. Vous pouvez employer d'autres méthodes pour modifier les valeurs des paramètres lors du lancement d'un programme Q Capture ou pendant son exécution. Toutefois, ces changements seront uniquement conservés dans la mémoire. Lors de l'arrêt et du redémarrage du programme Q Capture, les valeurs des paramètres qui ont été enregistrées dans la table IBMQREP_CAPPARMS sont utilisées. Vous pouvez mettre à jour cette table à l'aide du tableau de bord de réplication Q ou SQL.

Définition des valeurs de paramètres lors du démarrage

Lors du lancement d'un programme Q Capture, vous pouvez écraser les

valeurs des paramètres enregistrés dans la table `IBMQREP_CAPPARMS`. Vous pouvez utiliser le langage JCL ou la commande système `asnqcap` pour définir des valeurs pour les paramètres d'exploitation. Vos changements sont validés lors du lancement du programme, mais ne restent valides que pendant l'exécution du programme.

Modification dynamique de paramètres pendant l'exécution d'un programme Q Capture

Vous pouvez modifier les valeurs des paramètres de manière dynamique sans avoir à interrompre la capture des changements dans la source. Utilisez le paramètre `chgparms` de la commande `MODIFY` sous z/OS ou la commande `asnqccmd` sous Linux, UNIX, Windows ou des services système UNIX sous z/OS pour changer des valeurs tandis que le programme Q Capture est en cours d'exécution. Vos changements sont valides uniquement jusqu'à l'arrêt de l'exécution du programme ou jusqu'à la prochaine demande de changement des paramètres.

Trois modes de changement des paramètres d'exploitation de Q Capture

Supposons que vous souhaitiez diminuer le réglage par défaut pour le paramètre `commit_interval` de 500 millisecondes (une demi-seconde) pour un programme Q Capture qui est identifié par le schéma `ASN1` :

1. Mettez la table `IBMQREP_CAPPARMS` à jour pour le schéma `ASN1` de Q Capture. Définissez l'intervalle de validation à 250 millisecondes. Vous pouvez utiliser l'onglet Programmes dans le tableau de bord de réplication Q ou le langage SQL suivant :

```
update asn1.ibmqrep_capparms set commit_interval=250
```

Lors du prochain démarrage de ce programme Q Capture, l'intervalle de validation par défaut sera de 250 millisecondes.

2. Vous souhaitez connaître les effets d'un intervalle de validation plus long sur le rendement de réplication (nombre de transactions publiées pour une période donnée). Au lieu de modifier la valeur enregistrée dans la table de contrôle, démarrez le programme Q Capture avec `commit_interval` défini à 1 000 millisecondes (une seconde). Vous pouvez utiliser la commande `asnqcap` :

```
asnqcap capture_server=srddb1 capture_schema="ASN1" commit_interval=1000
```

Pendant que le programme s'exécute avec un intervalle de validation de 1 seconde, vous surveillez ses performances.

3. En fonction des performances, vous pouvez décider de réduire l'intervalle de validation. Au lieu d'arrêter le programme Q Capture, modifiez de manière dynamique le paramètre pendant l'exécution du programme à 750 millisecondes (0,75 seconde), et surveillez le changement. Vous pouvez utiliser le paramètre `chgparms` avec la commande `MODIFY` ou la commande `asnqccmd` :

```
f myqcap,chgparms commit_interval=750
```

```
asnqccmd capture_server=srddb1 capture_schema="ASN1" chgparms  
commit_interval=750
```

Vous pouvez continuer à surveiller les statistiques de rendement ou de latence et ajuster le paramètre `commit_interval`. Lorsque vous avez identifié la valeur répondant à vos besoins, vous pouvez mettre la table `IBMQREP_CAPPARMS` à jour (comme indiqué à l'étape 1). Lors du prochain démarrage d'un programme Q

Capture, la nouvelle valeur sera utilisée comme intervalle de validation par défaut.

Modification de paramètres pendant l'exécution d'un programme Q Capture

Vous pouvez modifier le comportement d'un programme Q Capture sans interrompre la capture des modifications depuis la source. Le programme Q Capture commence à utiliser immédiatement les nouveaux paramètres mais les modifications ne sont pas enregistrées dans la table de contrôle IBMQREP_CAPPARMS. Si vous arrêtez puis redémarrez le programme Q Apply, ce dernier utilise les valeurs sauvegardées dans la table de contrôle.

A propos de cette tâche

Vous pouvez modifier les paramètres suivants de Q Capture alors que ce programme est en cours d'exécution :

- **autostop**
- **commit_interval**
- **logreuse**
- **logstdout**
- **memory_limit**
- **monitor_interval**
- **monitor_limit**
- **prune_interval**
- **qfull_num_retries**
- **qfull_retry_delay**
- **signal_limit**
- **sleep_interval**
- **term**
- **trace_limit**

z/OS **Restriction** : La quantité de mémoire utilisée par le programme Q Capture est déterminée lors du démarrage du programme Q Capture, en fonction de la valeur du paramètre **memory_limit** et de la taille de la REGION spécifiée en JCL. Il est impossible de modifier la valeur du paramètre **memory_limit** lorsque le programme Q Capture est en cours d'exécution. Pour modifier cette valeur, vous devez préalablement arrêter le programme Q Capture.

Procédure

Pour changer dynamiquement des paramètres tandis que le programme Q Capture est en cours d'exécution, utilisez le paramètre **chgparms** avec la commande **MODIFY** sous z/OS ou la commande **asnqccmd** sous Linux, UNIX, Windows et les services système UNIX (USS) sous z/OS.

MODIFY

```
f myqcap,chgparms paramètre=valeur
```

Où myqcap est le nom du travail Q Capture.

asnqccmd

```
asnqccmd capture_server=serveur  
capture_schema=schéma  
chgparms paramètres
```

Où *serveur* représente le nom du serveur Q Capture, *schéma* identifie le programme Q Capture en cours d'exécution et *paramètres* désignent le(s) paramètre(s) que vous voulez modifier.

Lorsque vous modifiez les valeurs, un temps d'attente de 1 à 2 secondes peut apparaître entre le temps où vous émettez la commande et le temps où un programme Q Capture change son fonctionnement.

Changement des paramètres Q Capture sauvegardés dans la table IBMQREP_CAPPARMS

Un programme Q Capture stocke ses paramètres d'exploitation dans la table de contrôle IBMQREP_CAPPARMS. Pour modifier les valeurs des paramètres enregistrés, vous devez mettre la table de contrôle à jour.

A propos de cette tâche

Si vous écrasez ces paramètres enregistrés lors du démarrage du programme ou pendant son exécution, les modifications sont uniquement conservées dans la mémoire. La prochaine fois que vous démarrerez le programme Q Capture, il utilisera les valeurs sauvegardées dans la table de contrôle.

La table IBMQREP_CAPPARMS contient une seule ligne. Si cette table ne contient pas de lignes ou si elle contient plus d'une ligne, Q Capture ne fonctionnera pas.

Si vous souhaitez modifier une ou plusieurs valeurs de paramètres sauvegardés, vous pouvez mettre à jour la table IBMQREP_CAPPARMS. Comme un programme Q Capture lit cette table lorsqu'il démarre, vous devez arrêter et redémarrer le programme pour que les mises à jour prennent effet. La réinitialisation d'un programme Q Capture ne demandera pas au programme de lire de nouvelles valeurs dans la table IBMQREP_CAPPARMS.

Procédure

Pour modifier les paramètres Q Capture sauvegardés dans la table IBMQREP_CAPPARMS, appliquez l'une des méthodes suivantes :

Méthode	Description
Le tableau de bord de la réplication Q	Sous l'onglet Programmes, cliquez sur Propriétés puis sélectionnez un serveur Q Capture dans la zone Programme pour afficher ou modifier les valeurs sauvegardées.
SQL	A partir d'une invite de commande ou de l'un des outils de ligne de commande DB2, exécutez l'instruction SQL UPDATE pour la table IBMQREP_CAPPARMS. Par exemple, pour modifier les valeurs par défaut de monitor_interval et logstdout : <pre>update schéma.ibmqrep_capparms set monitor_interval=600, logstdout=Y</pre> où <i>schéma</i> identifie le programme Q Capture où se trouvent les valeurs des paramètres que vous souhaitez modifier.



Méthode	Description
Centre de réplication	Utilisez la fenêtre Modifier les paramètres sauvegardés. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture contenant le programme Q Capture dans lequel se trouvent les paramètres sauvegardés que vous souhaitez modifier ou voir affichés et sélectionnez Modifier les paramètres > Sauvegardés . Remarque : En démarrant avec la version 10.1, les nouveaux paramètres Q Capture n'ont pas tous été ajoutés au Centre de réplication. Pour afficher tous les paramètres, utilisez le tableau de bord de réplication Q.

Indication des transactions indésirables à ignorer par le programme Q Capture

Vous pouvez spécifier qu'un programme Q Capture ignore des transactions indésirables, et ces transactions ne seront pas capturées pour la réplication ou la publication.

A propos de cette tâche

Vous pouvez spécifier les transactions à ignorer en utilisant un ou plusieurs des identifiants suivants :

- ID de transaction
- ID d'autorisation
-  Clé d'accès
-  Nom de plan

Par exemple, votre opération peut exécuter des travaux de purge très importants une fois par mois et vous préférez exécuter ces travaux sur chaque base de données au lieu de répliquer les données. Dans ce cas, vous pourriez utiliser un ID spécifique et d'autorisation unique pour les travaux de purge, et indiquer que le Q Capture ignore les transactions à partir de cet ID.

Pour inviter le programme Q Capture pour ignorer une transaction unique basée sur son identificateur de transaction, vous utilisez un paramètre de commande **asnjcap** lorsque vous démarrez le programme Q Capture.

Pour ignorer les transactions en fonction d'un identificateur utilisateur, d'une clé d'accès ou d'un nom de plan, insérez ces identificateurs à l'aide d'un texte SQL dans la table de contrôle IBMQREP_IGNTRAN du serveur Q Capture. Vous pouvez utiliser un caractère générique (%) pour ignorer des groupes de transactions (voir ci-dessous pour plus de détails).

Attention : Le fait d'ignorer une transaction qui sera en dernier lieu avortée ne provoque pas de problèmes d'intégrité des données. Mais l'ignorance d'une transaction qui a été validée sur le serveur source, provoque généralement des divergences entre des serveurs source et cible, et il se peut que deviez prendre d'autres mesures pour synchroniser les serveurs, par exemple en déclenchant un nouveau chargement de la table cible ou en utilisant l'utilitaire de comparaison de table asntdiff.

Procédure

Pour inviter le programme Q Capture à ignorer les transactions :

1. Utilisez l'une des méthodes suivantes en fonction du plan de l'identifiant que vous prévoyez d'utiliser :

Identifiant	Procédure.
<p>ID de transaction</p>	<p>Le paramètre ignore_transid de la commande asnqcap permet de spécifier une transaction à ignorer. Le format de la commande est le suivant :</p> <pre>asnqcap capture_server=serveur_capture_q capture_schema=schéma_q_capture ignore_transid=ID_transaction</pre> <p>Sources DB2 <i>ID_transaction</i> est un identifiant hexadécimal de 10 octets sous le format suivant :</p> <p style="text-align: center;">z/OS</p> <p>0000:xxxx:xxxx:xxxx:mmmm</p> <p>Où <i>xxxx:xxxx:xxxx</i> correspond à l'ID de transaction et <i>mmmm</i> à l'ID membre pour le partage de données. Vous trouverez l'ID membre dans les deux derniers octets de l'en-tête de l'enregistrement de journal, dans la sortie LOGP. L'ID membre est 0000 si le partage de données n'est pas activé.</p> <p>Par exemple, la commande suivante spécifie qu'une transaction doit être ignorée au sein d'un environnement de partage des données avec un ID membre 0001 :</p> <pre>asnqcap capture_server=sample capture_schema=ASN ignore_transid =0000:BD71:1E23:B089:0001</pre> <p style="text-align: center;">Linux UNIX Windows</p> <p>Version 9.7 et précédentes <i>nnnn:0000:xxxx:xxxx:xxxx</i></p> <p>Où <i>xxxx:xxxx:xxxx</i> représente l'ID de transaction et <i>nnnn</i>, l'identificateur de partition pour les bases de données partitionnées (0000 pour les bases de données non partitionnées).</p> <p>Par exemple, la commande suivante spécifie qu'une transaction doit être ignorée au sein d'une base de données non partitionnée :</p> <pre>asnqcap capture_server=sample capture_schema=ASN ignore_transid =0000:0000:0000:0000:BE97</pre>

Identifiant	Procédure.
ID de transaction (suite)	<p>Version 10.1 ou supérieure avec le paramètre compatibility de 1001 ou supérieur, ou version 9.8 <code>0000:llll:xxxx:xxxx:xxxx</code></p> <p>Où <code>xxxx:xxxx:xxxx</code> est l'ID transaction et <code>llll</code> est l'identificateur de flux de journalisation avec la fonction DB2 pureScale. Les quatre premiers caractères sont toujours 0000.</p> <p>Par exemple, la commande suivante indique qu'une transaction doit être ignorée dans une base de données avec la fonction DB2 pureScale :</p> <pre>asnqcap capture_server=sample capture_schema=ASN ignore_transid =0000:0001:0000:0000:BE97</pre> <p>sources Oracle L'ID de transaction est la valeur XID de la vue V\$CONTENTS de la fonctionnalité Oracle Log Miner. L'ID est une valeur RAW(8). Lorsqu'il est affiché dans le texte, l'ID de transaction est formaté en tant que chaîne hexadécimale (16 chiffres au total).</p> <p>Pour ignorer plusieurs transactions, arrêtez le programme Q Capture puis redémarrez-le à chaud en spécifiant un autre identificateur de transaction.</p>

Identifiant	Procédure.
Identificateur d'autorisation, clé d'accès (z/OS) ou nom de plan (z/OS)	<p>Utilisez SQL pour insérer un ou plusieurs identificateurs dans la table de contrôle IBMQREP_IGNTRAN. Insérez les identificateurs appropriés dans les colonnes suivantes :</p> <p>AUTHID ID d'autorisation. sources Oracle : L'ID d'autorisation est la valeur USERNAME de la vue V\$CONTENTS de la fonctionnalité Oracle Log Miner.</p> <p>z/OS AUTHTOKEN Jeton d'autorisation (nom de travail).</p> <p>z/OS PLANNAME Nom de plan.</p> <p>Par exemple, l'instruction suivante indique que le programme Q Capture doit ignorer une transaction dont l'identificateur utilisateur est repldba :</p> <pre>insert into schéma.IBMQREP_IGNTRAN (AUTHID, AUTHTOKEN, PLANNAME) values ('repldba', NULL, NULL);</pre> <p>Vous pouvez utiliser un caractère générique, le signe pourcentage (%), pour représenter n'importe quel nombre de caractères ou aucun, ce qui vous permet d'ignorer des groupes d'ID autorisation, de clés d'accès ou de noms de plan.</p> <p>Par exemple, l'instruction d'insertion suivante demande à Q Capture d'ignorer toute transaction ayant des noms de plan qui se terminent par la chaîne "PLAN11" :</p> <pre>INSERT INTO IBMQREP_IGNTRAN (AUTHID, AUTHTOKEN, PLANNAME, IGNTRANTRC) VALUES (null, null, '%PLAN11', 'N')</pre> <p>Cet exemple indique à Q Capture d'ignorer toute transaction avec un ID d'autorisation qui commence par "AUTH" et contient la sous-chaîne "ID". De la même manière, la clé d'accès doit correspondre à une chaîne commençant par "TOK11" (z/OS uniquement) :</p> <pre>INSERT INTO IBMQREP_IGNTRAN (AUTHID, AUTHTOKEN, PLANNAME, IGNTRANTRC) VALUES ('AUTH%ID%', 'TOK11%', null, 'N')</pre> <p>Si le signe pourcentage fait partie des données, placez une barre oblique inversée (\) devant. Par exemple, cette instruction indique que Q Capture ignore l'ID d'autorisation AUTH%ID :</p> <pre>INSERT INTO IBMQREP_IGNTRAN (AUTHID, AUTHTOKEN, PLANNAME, IGNTRANTRC) VALUES ('AUTH\%ID', null, null, 'N')</pre> <p>Les masques de correspondance sont sensibles à la casse.</p>

2. **Facultatif** : Indiquez si vous voulez que le programme Q Capture insère une ligne dans la table IBMQREP_IGNTRANTRC lorsqu'il ignore une transaction.

Par défaut, cette fonction de trace est désactivée. Pour activer la fonction de trace, utilisez l'instruction SQL suivante :

```
mettre à jour schéma.IBMQREP_IGNTRAN set IGNTRANTRC='Y' OU  
identificateur = identificateur_transaction
```

L'*identificateur* serait l'une des colonnes AUTHID, AUTHTOKEN, ou PLANNAME qui étaient utilisées afin d'identifier la ou les transactions à ignorer.

Lorsque vous utilisez le paramètre Q Apply **insert_bidi_signal**=n, le paramétrage de IGNTRANTRC sur N (pas de fonction de trace) empêche le programme Q Capture d'insérer une ligne dans la table IBMQREP_IGNTRANTRC pour chaque transaction qu'il ne recapture pas et réduit les frais de maintenance sur la table.

Arrêt d'un programme Q Capture

Vous pouvez arrêter un programme Q Capture et il cessera alors de lire à partir du journal de récupération et de construire des transactions dans la mémoire.

A propos de cette tâche

Lorsque vous arrêtez Q Capture, les messages placés dans les files d'attente seront validés dans WebSphere MQ avant l'arrêt du programme. Les transactions WebSphere MQ ou changements de ligne non validés qui étaient dans la mémoire au moment où vous avez arrêté le programme seront recapturés à partir du journal lorsque le programme Q Capture redémarre, selon un point de reprise stocké dans le message de redémarrage.

Conseil : Vous n'avez pas besoin d'arrêter un programme Q Capture pour ajouter ou supprimer un abonnement Q ou une publication :

- Si vous souhaitez ajouter un ou deux abonnements Q ou publications pendant que le programme est exécuté, créez les abonnements Q ou publications pour qu'ils ne démarrent pas automatiquement, puis activez-les.
- Si vous souhaitez ajouter un grand nombre d'abonnements Q ou de publications, créez-les afin qu'ils démarrent automatiquement, puis réinitialisez le programme Q Capture.
- Vous pouvez supprimer un abonnement Q ou une publication sans arrêter le programme Q Capture en désactivant l'abonnement Q ou la publication, puis supprimez-le.

Procédure

Pour arrêter un programme Q Capture, appliquez l'une des méthodes suivantes :

Méthode	Description
Paramètre stop	<p>Utilisez ce paramètre avec la commande MODIFY sous z/OS ou la commande asnqccmd sous Linux, UNIX, Windows et UNIX System Services sous z/OS pour arrêter un programme Q Capture :</p> <p>MODIFY f myqcap,stop</p> <p>Où myqcap est le nom du travail Q Capture.</p> <p>asnqccmd asnqccmd capture_server=<i>nom_serveur</i> capture_schema=<i>schéma</i> stop</p> <p>Où <i>nom_serveur</i> est le nom de la base de données ou sous-système où le programme Q Capture est en cours d'exécution et <i>schéma</i> identifie le programme Q Capture que vous souhaitez arrêter.</p>
SQL	<p>Utilisez une invite de commande ou un des outils de ligne de commande DB2 pour insérer un signal STOP dans la table IBMQREP_SIGNAL au niveau du serveur Q Capture :</p> <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT TIMESTAMP, 'CMD', 'STOP', 'NULL', 'P');</pre> <p>Où schéma identifie le programme Q Capture que vous souhaitez arrêter.</p>
<div style="background-color: #800000; color: white; padding: 2px; text-align: center; font-weight: bold;">Windows</div> Services Windows	<p>Vous pouvez créer un service de réplication DB2 sur le système d'exploitation Windows et utilisez le gestionnaire de contrôle de service Windows ou la commande net stop pour arrêter un programme Q Capture.</p>

Arrêt d'un programme Q Capture à un point spécifié

Vous pouvez effectuer un arrêt planifié et contrôlé du programme Q Capture, par exemple lorsque vous stoppez un site lors d'un arrêt planifié.

A propos de cette tâche

L'arrêt possible de Q Capture à un point spécifié garantit la livraison ou l'application de toutes les modifications à une cible avant que vous arrêtez la réplication. Cette option est utile dans les cas où vous redirigez la charge de travail de réplication d'un site actif vers un site de secours à un moment précis.

La commande **stop** et le signal STOP de Q Capture offrent deux paramètres utilisables pour un arrêt contrôlé :

captureupto

Le paramètre **captureupto** indique un point temporel où Q Capture arrêtera le journal de récupération. Vous pouvez indiquer un horodatage

dans le fuseau horaire de la base de données. Q Capture arrête de publier toutes les transactions validées avant ce moment. Vous pouvez également indiquer la valeur `CURRENT_TIMESTAMP` pour ordonner à Q Capture d'arrêter la lecture du journal à l'heure locale actuelle. Vous pouvez également indiquer `EOL` pour que Q Capture s'arrête à la fin du journal.

stopafter

Si vous indiquez le paramètre **stopafter**, Q Capture suspend la lecture du journal jusqu'à ce que l'une des deux conditions indiquées soit vraie :

data_applied

Q Capture s'arrête après que toutes les modifications situées après le point d'arrêt spécifié sont appliquées à la cible. Les applications peuvent alors être redirigées du site en panne vers le site de basculement.

data_sent

Q Capture s'arrête après la suppression de tous les messages de la file d'attente de transmission pointant vers le système cible ou après le traitement de tous les messages par le programme Q Apply dans le cas d'une file d'attente de réception/d'envoi locale partagée. Pour les configurations distantes, cette méthode garantit que tous les messages publiés n'existent que dans la file d'attente locale sur le système cible. Vous pouvez alors effectuer les tâches de maintenance telles que l'arrêt de WebSphere MQ sur le site en panne sans risquer de perdre des messages. Ce mode présente l'avantage de fonctionner même si Q Apply ne lit pas activement le contenu des files d'attente (bien que l'espace disque doive être suffisant sur le système de basculement pour que tous les messages soient livrés).

Vous pouvez indiquer **captureupto** ou **stopafter**, voire les deux.

Cet horodatage doit être spécifié en tant que format d'horodatage partiel ou complet pour le fuseau horaire du serveur Q Capture. L'horodatage complet utilise le format suivant : `YYYY-MM-DD-HH.MM.SS.mmmmmm`. Par exemple, `2010-04-10-10.35.30.555555` est l'horodatage local pour le 10 avril 2010, 10 h 35, 30 secondes et 55 555 microsecondes. Vous pouvez spécifier l'horodatage partiel en utilisant l'un des formats suivants :

`YYYY-MM-DD-HH.MM.SS`

Par exemple, `2010-04-10-23.35.30` correspond à l'horodatage local partiel pour le 10 avril 2010 à 23 h 35 et 30 secondes.

`YYYY-MM-DD-HH.MM`

Par exemple, `2010-04-10-13.30` correspond à l'horodatage local partiel pour le 10 avril 2010 à 13 h 30

`YYYY-MM-DD-HH`

Par exemple, `2010-04-10-01` correspond à l'horodatage local partiel pour le 10 avril 2010 à 1 heure

`HH.MM`

Par exemple, `14:55` correspond à l'horodatage local partiel de la journée en cours à 14 heures et 55 minutes.

`HH`

Par exemple, `14` correspond à l'horodatage local partiel de la journée en cours à 14 heures.

Remarque : Si vous indiquez le paramètre **data_sent** et que Q Capture détecte qu'une autre application partage la file d'attente de transmission, le programme émet un message d'avertissement et s'arrête car il ne peut déterminer avec certitude quand tous ses messages seront supprimés de la file d'attente.

Procédure

Pour arrêter un programme Q Capture à un point indiqué, appliquez l'une des méthodes suivantes :

Méthode	Description
<p>Commande stop avec l'option captureupto et/ou l'option stopafter</p>	<p>Utilisez ces paramètres avec la commande MODIFY sous z/OS ou la commande asqccmd sous Linux, UNIX, Windows et UNIX System Services sous z/OS pour arrêter un programme Q Capture à un point spécifié :</p> <p>MODIFY</p> <p>Le format de la commande se présente ainsi :</p> <pre>F Q_CAPTURE_JOBNAME,STOP CAPTUREUPTO=<i>timestamp</i> CURRENT_TIMESTAMP EOL STOPAFTER=DATA_SENT DATA_APPLIED</pre> <p>Où <i>Q_CAPTURE_JOBNAME</i> est le nom du travail Q Capture.</p> <p>asqccmd</p> <p>Le format de la commande se présente ainsi :</p> <pre>asqccmd capture_server=<i>nom_serveur</i> capture_schema=<i>schéma</i> stop captureupto=<i>horodatage</i> CURRENT_TIMESTAMP EOL stopafter=<i>data_sent</i> <i>data_applied</i></pre> <p>Où <i>nom_serveur</i> est le nom de la base de données ou sous-système où le programme Q Capture est en cours d'exécution et <i>schéma</i> identifie le programme Q Capture que vous souhaitez arrêter.</p> <p>Les exemples suivants proposent différentes façons d'utiliser les paramètres controlled-stop :</p> <pre>f QCAP1,STOP CAPTUREUPTO=2010-04-05-22:30:00.000000 asqccmd capture_server=sample capture_schema="bsn" stop captureupto=2010-04-05-22:30:00.000000</pre> <p>Q Capture publie toutes les modifications validées jusqu'au 5 avril 2010 à 22 h 30 inclus.</p> <pre>f QCAP1,STOP STOPAFTER=DATA_SENT asqccmd capture_server=sample capture_schema="bsn" stop stopafter=data_sent</pre> <p>Q Capture arrête de lire le journal immédiatement et attend que tous les messages aient été envoyés de la file d'attente de transmission ou de la file d'attente d'envoi et de réception locale avant l'arrêt du programme.</p> <pre>f QCAP1,STOP CAPTUREUPTO=2010-04-05-22:30:00.000000 STOPAFTER=DATA_APPLIED asqccmd capture_server=sample capture_schema="bsn" stop captureupto=2010-04-05-22:30:00.000000 stopafter=data_applied</pre> <p>Q Capture publie toutes les modifications validées jusqu'au 4 avril 2010 à 22 h 30 inclus, et il attend que Q Apply finisse s'appliquer toutes les données avant leur publication jusqu'à ce point temporel, puis il s'arrête.</p>

Méthode	Description
Signal STOP avec l'option captureupto option et/ou l'option stop after	<p>Utilisez une invite de commande ou un des outils de ligne de commande DB2 pour insérer un signal STOP avec un ou plusieurs paramètres controlled-stop dans la table IBMQREP_SIGNAL au niveau du serveur Q Capture. Le format du signal se présente ainsi :</p> <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'STOP', 'horodatage CURRENT_TIMESTAMP EOL;DATA_SENT DATA_APPLIED', 'P');</pre> <p>Où schema identifie le programme Q Capture à arrêter et où <i>timestamp</i> indique l'heure à laquelle Q Capture doit arrêter la lecture du journal, puis s'arrêter (ou s'arrêter lorsqu'une condition est remplie).</p>

Quand Q Capture attend une réponse de Q Apply ou le nettoyage de la file d'attente de transmission, il passe à l'état suspendu et voici ce qui se passe :

- Un message d'information est émis.
- L'exécution du lecteur de journal est mise en suspens.
- Les messages de contact sont toujours envoyés.
- Les messages d'administration sont toujours traités.
- Les statistiques sont toujours écrites sur les tables du moniteur.
- Q Capture continue à répondre aux commandes MODIFY ou asnqccmd, y compris à signaler l'état suspendu d'une commande **status**.

L'interruption d'un programme Q Capture mis en suspend requiert la commande **stop** par défaut sans paramètre ou un signal du système d'exploitation (par exemple, *f nom_travail stop* sous z/OS ou Ctrl+C sous Linux, UNIX et Windows). Les signaux envoyés à la table IBMQREP_SIGNALS pour annuler l'arrêt contrôlé ne fonctionnent pas car le lecteur de journal est mis en suspens.

Démarrage des abonnements Q

Vous démarrez des abonnements Q pour indiquer au programme Q Capture qu'il peut commencer à capturer les changements apportés aux tables source et placer les messages de changements dans les files d'attentes de WebSphere MQ.

A propos de cette tâche

Avant que le programme Q Capture puisse démarrer la réplication des données à partir des tables source, les abonnements Q définissant ces tables source doivent être à l'état A (actif) ou N (nouveau). Par défaut, les abonnements Q venant d'être créés sont à l'état N et sont démarrés automatiquement au lancement ou à la réinitialisation du programme Q Capture. Procédez comme suit pour démarrer des abonnements Q existants et les placer dans l'état A (actif).

Le programme Q Capture doit être en cours d'exécution pour lire le signal CAPSTART. Si le programme Q Capture est arrêté lorsque vous démarrez des abonnements Q, le programme ne traite le signal que s'il s'agit d'un démarrage à chaud. Le signal ou le message sera perdu si vous utilisez le démarrage à froid.

Si l'un de vos abonnements Q que vous souhaitez démarrer définit un chargement automatique utilisant l'utilitaire EXPORT, vous devez créer un fichier de mots de passe sur le serveur Q Apply afin d'autoriser l'utilitaire à se connecter au serveur Q Capture.

Procédure

Pour démarrer les abonnements Q, appliquez une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande START QSUB. Par exemple, les commandes suivantes définissent l'environnement et génèrent SQL pour démarrer la description Q DEPARTMENT0001 : ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE ASN1; START QSUB SUBNAME DEPARTMENT0001;
Le tableau de bord de la réplication Q	Dans l'onglet Abonnements, sélectionnez un ou plusieurs abonnements Q et cliquez sur Démarrer .
Centre de réplication	Utilisez la fenêtre Gérer les abonnements Q. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture où la table source de l'abonnement Q est localisée puis sélectionnez Gérer > Abonnements Q .
SQL	Utilisez une invite de commande ou un des outils de ligne de commande DB2 pour insérer un signal CAPSTART dans la table IBMQREP_SIGNAL au niveau du serveur Q Capture : insert into <i>schéma</i> .IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT TIMESTAMP, 'CMD', 'CAPSTART', ' <i>sous-nom</i> ', 'P'); où <i>schéma</i> identifie un programme Q Capture et <i>sous-nom</i> est le nom de l'abonnement Q que vous souhaitez démarrer.

Arrêt des abonnements Q

Vous arrêtez un abonnement Q pour charger le programme Q Capture d'arrêter la capture des modifications de l'abonnement Q. L'arrêt vous permet de supprimer ou de suspendre l'activité des abonnements Q sans arrêter le programme Q Capture.

Avant de commencer

Le programme Q Capture doit être exécuté pour lire le signal CAPSTOP. Si le programme Q Capture est arrêté lorsque vous arrêtez un abonnement Q, le programme ne traitera le signal que s'il est démarré à chaud. Le signal sera perdu si vous utilisez le démarrage à froid.

A propos de cette tâche

Lorsque vous arrêtez un abonnement Q, le programme Q Capture arrête la capture des modifications de l'abonnement Q et fait passer son état sur I (inactif) dans la table IBMQREP_SUBS.

Procédure

Pour arrêter les abonnements Q, appliquez une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande STOP QSUB. Par exemple, les commandes suivantes définissent l'environnement et génèrent SQL pour arrêter la description Q DEPARTMENT0001 : ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE ASN1; STOP QSUB SUBNAME DEPARTMENT0001;
Le tableau de bord de la réplication Q	Dans l'onglet Abonnements, sélectionnez un ou plusieurs abonnements Q et cliquez sur Arrêter .
Centre de réplication	Utilisez la fenêtre Gérer les abonnements Q. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture où la table source de l'abonnement Q est localisée puis sélectionnez Gérer > Abonnements Q .
SQL	Utilisez une invite de commande ou un des outils de ligne de commande DB2 pour insérer un signal CAPSTOP dans la table IBMQREP_SIGNAL au niveau du serveur Q Capture : <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT TIMESTAMP, 'CMD', 'CAPSTOP', 'sous-nom', 'P');</pre> où <i>schéma</i> identifie un programme Q Capture et <i>sous-nom</i> est le nom de l'abonnement Q que vous souhaitez arrêter.

Démarrage des publications

Lorsque vous démarrez des publications XML, le programme Q Capture lance la capture des modifications source et met les messages de modification dans les files d'attente de WebSphere MQ.

Avant de commencer

- Le programme Q Capture doit être exécuté pour lire le signal CAPSTART ou le message activer l'abonnement. Si le programme Q Capture est arrêté lorsque vous démarrez des publications XML, il ne traitera le signal ou le message qu'après un démarrage à chaud. Le signal ou le message sera perdu si vous utilisez le démarrage à froid.
- Pour que le programme Q Capture puisse commencer à publier des modifications à partir des tables source, les publications XML qui définissent ces tables doivent être à l'état A (actif) ou N (nouveau).

A propos de cette tâche

Par défaut, les publications XML venant d'être créées sont à l'état N (nouveau) et sont automatiquement démarrées lorsque le programme Q Capture est démarré ou réinitialisé. Si vous modifiez cette option par défaut, vous devez démarrer les publications XML après les avoir créées.

Procédure

Pour démarrer des publications XML, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande START PUB . Par exemple, les commandes suivantes définissent l'environnement et génèrent le code SQL nécessaire pour démarrer la publication EMPLOYEE0001 : ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE EP1; START PUB PUBNAME EMPLOYEE0001;
Centre de réplcation	Utilisez la fenêtre Gestion des publications. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture où se trouve la table source de la publication XML, puis sélectionnez Gestion > des publications XML .

Méthode	Description
SQL	<p>Utilisez une invite de commande ou l'un des outils de ligne de commande DB2 pour insérer un signal CAPSTART dans la table IBMQREP_SIGNAL sur le serveur Q Capture :</p> <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'CAPSTART', 'nompub', 'P');</pre> <p>Où <i>schéma</i> identifie un programme Q Capture et <i>nompub</i> correspond au nom de la publication XML que vous souhaitez démarrer.</p>

Arrêt des publications

Vous arrêtez une publication XML pour charger le programme Q Capture d'arrêter la capture des modifications de la publication. Cette fonction permet de supprimer ou de suspendre l'activité des publications sans arrêter le programme Q Capture.

Avant de commencer

Le programme Q Capture doit être exécuté pour lire le signal CAPSTOP. Si le programme Q Capture est arrêté lorsque vous arrêtez une publication, il ne traitera le signal qu'après un démarrage à chaud. Le signal sera perdu si vous utilisez le démarrage à froid.

A propos de cette tâche

Lorsque vous arrêtez une publication, le programme Q Capture arrête la capture des modifications de la publication XML et fait passer son état sur I (inactif) dans la table IBMQREP_SUBS.

Procédure

Pour arrêter des publications XML, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande STOP PUB. Par exemple, les commandes suivantes définissent l'environnement et génèrent le code SQL nécessaire pour arrêter la publication EMPLOYEE0001 :</p> <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE EP1;</pre> <p>STOP PUB PUBNAME EMPLOYEE0001;</p>
Centre de répliation	<p>Utilisez la fenêtre Gestion des publications. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture où se trouve la table source de la publication XML, puis sélectionnez Gestion > des publications XML.</p>

Méthode	Description
SQL	<p>Utilisez une invite de commande ou l'un des outils de ligne de commande DB2 pour insérer un signal CAPSTOP dans la table IBMQREP_SIGNAL sur le serveur Q Capture :</p> <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'CAPSTOP', 'nompub', 'P');</pre> <p>Où <i>schéma</i> identifie un programme Q Capture et <i>nompub</i> correspond au nom de la publication XML que vous souhaitez désactiver.</p>

Gestion de l'activité des messages Q Capture au niveau de la file d'attente d'envoi

Vous pouvez gérer l'activité de messages Q Capture au niveau de la file d'attente d'envoi pour éviter les indisponibilités lorsqu'une erreur se produit sur une file d'attente, pour éviter de devoir effectuer une régénération intégrale (nouveau chargement) des tables répliquées en raison d'une erreur de file d'attente, et pour garantir la livraison des événements publiés.

Le Q Capture programme Q Capture gère les informations de redémarrage individuel pour chaque file d'attente d'envoi. Si une file d'attente s'arrête en raison d'une erreur ou d'une maintenance planifiée, Q Capture peut réeffectuer les modifications qui se sont produites pendant que la file d'attente était arrêtée. Grâce à cette fonction, il n'est pas nécessaire de recharger les tables cible utilisant une file d'attente désactivée.

Vous pouvez également définir le programme Q Capture pour qu'il continue à insérer des messages dans des files d'attente d'envoi lorsqu'une ou plusieurs files d'attente sont désactivées. Q Capture fournit des commandes permettant d'arrêter d'insérer des messages dans des files d'attente d'envoi individuelles pour que vous puissiez résoudre des incidents ou effectuer une maintenance, et permettant de redémarrer les files d'attente d'envoi sélectionnées lorsque vous êtes prêt à reprendre la réplication ou la publication.

Les sections suivantes fournissent plus d'informations sur les fonctions Q Capture vous permettant de gérer des files d'attente d'envoi individuelles :

- «Utilisation de files d'attente d'envoi multiples», à la page 274
- «Définition de l'action d'erreur pour les files d'attente d'envoi sur Q», à la page 274
- «Garder les files d'attente désactivées inactives lors du démarrage de Q Capture», à la page 275
- «Informations de file d'attente d'envoi dans le message de redémarrage», à la page 275

Utilisation de files d'attente d'envoi multiples

Généralement, vous configurez Q Capture pour répartir les messages sur plusieurs files d'attente d'envoi pour deux raisons :

Répartition de la charge de travail par application

Les modifications des tables correspondant à chaque application sont répliquées indépendamment des modifications des autres tables d'application, ce qui réduit le temps système d'administration et de l'unité centrale.

Distribution des données

Un programme Q Capture unique capture les modifications une fois, puis les enregistre dans chaque file d'attente d'envoi pour qu'elles soient livrées au serveur distant associé.

La figure 28 affiche une configuration de réplication unidirectionnelle dans laquelle les modifications sont réparties entre quatre files d'attente d'envoi, chaque file d'attente gérant l'activité d'une table source d'une application distincte.

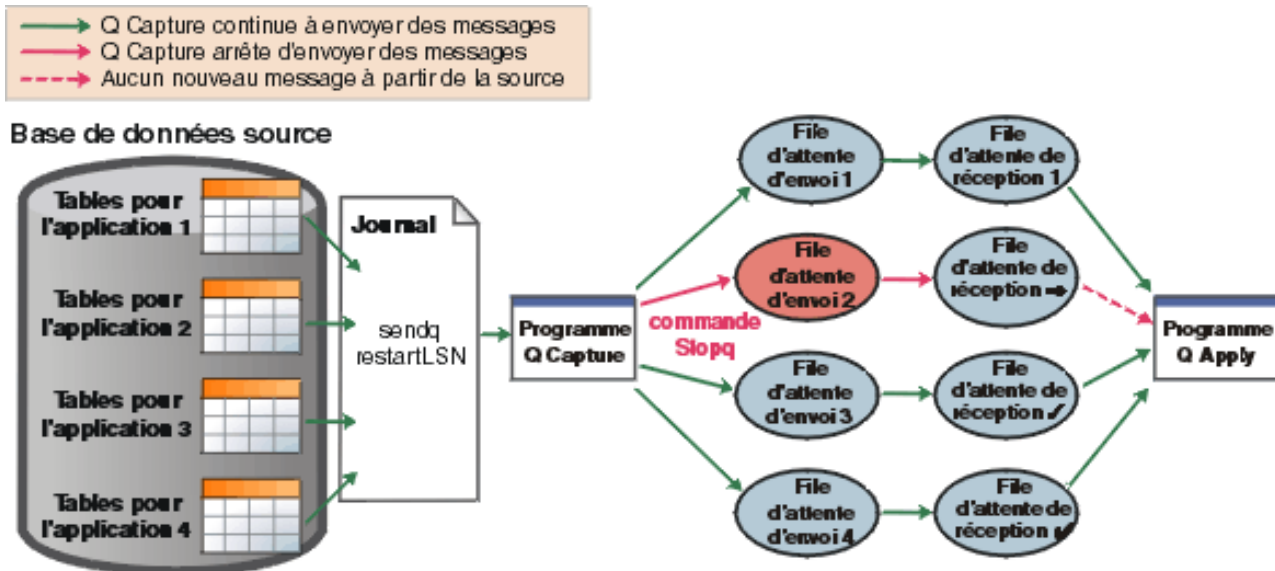


Figure 28. Gestion de l'activité de messages d'une file d'attente d'envoi unique. La commande stopq a été émise pour la file d'attente d'envoi 2, qui indique au programme Q Capture d'arrêter d'insérer des messages dans la file d'attente. Lorsque l'activité de la source est arrêtée, vous pouvez effectuer une maintenance sur les tables de l'application 2 sans arrêter la réplication pour les 1, 3 et 4. Une fois la maintenance terminée, vous pouvez émettre la commande startq et Q Capture recherche le point de redémarrage de la file d'attente d'envoi 2 (sendqrestartLSN) dans le journal de récupération et commence la capture des modifications pour la file d'attente d'envoi maintenant active, à partir de ce point.

Définition de l'action d'erreur pour les files d'attente d'envoi sur Q

Pour empêcher l'échec d'une file d'attente d'envoi d'affecter la réplication ou la publication sur les autres files d'attente, mappez les files d'attente sur Q (**Stop send queue** dans le Centre de réplication). Avec cette action d'erreur, lorsqu'une file d'attente d'envoi est désactivée (parce qu'elle n'a plus d'espace disponible pour les messages), le programme Q Capture arrête d'insérer des messages dans la file d'attente désactivée mais continue à insérer des messages dans les autres files d'attente.

Une fois l'incident résolu sur le file d'attente d'envoi désactivée, vous pouvez utiliser la commande **startq** pour indiquer à Q Capture de reprendre la lecture du journal de récupération pour les publications ou les abonnements Q utilisant la file d'attente d'envoi et de reprendre l'insertion de messages dans la file d'attente.

Garder les files d'attente désactivées inactives lors du démarrage de Q Capture

Par défaut, le programme Q Capture active toutes les files d'attente d'envoi inactives lorsqu'il démarre ou lorsque vous utilisez la commande Q Capture **reinit** pour recharger tous les abonnements Q à partir des tables de contrôle Q Capture. Si vous devez garder toutes les files d'attente d'envoi désactivées à l'état inactif (I), vous pouvez démarrer Q Capture avec le paramètre **startallq** défini sur N.

Informations de file d'attente d'envoi dans le message de redémarrage

Q Capture conserve un message de redémarrage sur sa file d'attente de redémarrage pour effectuer le suivi des points de redémarrage dans le journal de récupération. Depuis la version 9.5, groupe de correctifs 3, le message contient des informations pour chaque file d'attente d'envoi.

Important : Si votre version du programme Q Capture est la version 9.5 avec le groupe de correctifs 3 ou une version ultérieure (sous z/OS, il s'agit de la version 9.1 avec la modification provisoire du logiciel correspondant à la version 9.5 avec le groupe de correctifs 3), le programme peut démarrer en mode à chaud en utilisant les informations de démarrage d'un niveau plus ancien de Q Capture. En revanche, les programmes Q Capture antérieurs à la version 9.5 groupe de correctifs 3, ne peuvent pas utiliser le nouveau format de message de redémarrage pour redémarrer en mode à chaud. Si vous devez redémarrer un ancien niveau de Q Capture en mode à chaud, vous devez démarrer Q Capture à partir d'un emplacement connu du journal en définissant les paramètres **1sn** et **maxcmtseq**. Recherchez le message ASN7109I dans le journal de diagnostic Q Capture. Pour plus d'informations, voir «Démarrage de Q Capture à partir d'un emplacement connu dans le journal DB2», à la page 249.

Vous pouvez mettre en forme le nouveau message de redémarrage en utilisant la commande **asnqfmt** ou la commande **asnqxfmt**. L'exemple suivant illustre un message de redémarrage contenant des informations de redémarrage pour trois files d'attente d'envoi : Q1, Q2 et Q3.

```
**** Descriptor for message number: 1
  StrucId      MD
  MsgId        414d5120535a5051544553542020202048c1cd6620004e02
  CorrelId     51505542214d41494e4c494e450000000000000000000000
  Version      1
  Report       0
  MsgType      8
  Expiry       -1
  Feedback     0
  Encoding     273
  CodedCharSetId 819
  Format        COMPACT
  Priority      0
  Persistence  1
**** Message number: 1
**** Message size: 640
  qRestartMsg for MAINLINE log reader.
  qRestartMsg.capServer : QTEST
  qRestartMsg.capSchema : ASN
```

```

qRestartMsg.qRestartsgSize : 640
Number of partition at restart: 1
restartLSN=0000:0000:0000:1e46:2be0 nodeId=0
qRestartMsg.lastCommitSEQ: 0000:0000:0000:0000:0000
qRestartMsg.lastCommitTime: 2008-09-11-17.22.42.287478
qRestartMsg.reuseSEQ: 0000:0000:0000:0000:0000
qRestartMsg.reuseTime: 2008-09-11-17.22.42
Number of send queues at restart: 3
[0] sendq name: Q2
    sendq activation time: 1221178963
    sendq next msg seq no: 00000000000000000000000000000003
    sendq lastCommitSEQ: 0000:0000:0000:0000:0000
    sendq lastCommitTime: 2008-09-11-17.22.42.287478
    sendq restartLSN=0000:0000:0000:1e46:2be0 nodeId=0
[1] sendq name: Q1
    sendq activation time: 1221178963
    sendq next msg seq no: 00000000000000000000000000000003
    sendq lastCommitSEQ: 0000:0000:0000:0000:0000
    sendq lastCommitTime: 2008-09-11-17.22.42.287478
    sendq restartLSN=0000:0000:0000:1e46:2be0 nodeId=0
[2] sendq name: Q3
    sendq activation time: 1221178963
    sendq next msg seq no: 00000000000000000000000000000003
    sendq lastCommitSEQ: 0000:0000:0000:0000:0000
    sendq lastCommitTime: 2008-09-11-17.22.42.287478
    sendq restartLSN=0000:0000:0000:1e46:2be0 nodeId=0

```

Démarrage de l'activité de message sur une ou plusieurs files d'attente d'envoi

Vous pouvez indiquer au programme Q Capture de reprendre l'insertion de messages sur une ou plusieurs files d'attente d'envoi inactives sans devoir effectuer une régénération intégrale (nouveau chargement) pour les tables cible utilisant la file d'attente.

A propos de cette tâche

Il est utile de pouvoir démarrer une ou plusieurs files d'attente d'envoi lorsque vous définissez l'action erreur pour le mappage de file d'attente de réplication ou de publication sur Q, qui indique au programme Q Capture de continuer à insérer des messages dans les files d'attente actives si une erreur se produit sur une ou plusieurs autres files d'attente d'envoi.

Lorsque vous démarrez l'activité de messages sur une plusieurs files d'attente d'envoi, Q Capture recherche le point de redémarrage dans le journal de récupération pour toutes les files d'attente arrêtées et il peut ainsi leur faire rattraper les files d'attente actives. Comme Q Capture conserve les abonnements Q actifs lorsque les files d'attente sont arrêtées, il n'est pas nécessaire d'effectuer une régénération intégrale (chargement) des tables cible lors les files d'attente sont redémarrées.

Remarque : Lorsque Q Capture met à jour les files d'attente redémarrées, le temps d'attente global est affecté car Q Capture n'insère pas de messages dans les autres files d'attente tant qu'il ne lit pas le journal au-delà du point de validation maximal pour ces files d'attente déjà actives.

Procédure

Pour démarrer une ou plusieurs files d'attente d'envoi, utilisez l'une des méthodes suivantes :

Méthode	Description
Commande startq	<p>Vous pouvez utiliser la commande z/OS MODIFY ou la commande asnqcmd pour émettre startq. La commande vous permet de démarrer une file d'attente d'envoi unique ou toutes les files d'attente qu'utilise un programme Q Capture.</p> <p>Dans les exemples ci-dessous, la première commande MODIFY indique au programme Q Capture de reprendre l'insertion de messages dans la file d'attente d'envoi <code>asn.qm1.dataq</code>. La deuxième commande redémarre toutes les files d'attente :</p> <pre>f myqcap,startq=asn.qm1.dataq</pre> <pre>f myqcap,startq=all</pre> <p>Les commandes suivantes utilisent asnqcmd pour les mêmes actions :</p> <pre>asnqcmd capture_server=sourcedb capture_schema="asn" startq="asn.qm1.dataq"</pre> <pre>asnqcmd capture_server=sourcedb capture_schema="asn" startq=all</pre>
Réinitialisation du programme Q Capture	<p>La commande reinit de Q Capture démarre toute les files d'attente d'envoi inactives si le programme Q Capture a été démarré avec l'option startallq=y (paramètre par défaut).</p>
Arrêt et démarrage du programme Q Capture	<p>Lorsque Q Capture démarre, il réinitialise par défaut toutes les files d'attente d'envoi sur l'état actif (A). Arrêter et démarrer Q Capture indique également au programme de régénérer toutes les publications ou tous les abonnements Q dans ses tables de contrôle et de rechercher toutes les modifications effectuées.</p> <p>Vous pouvez démarrer Q avec le paramètre startallq=n pour conserver toutes les files d'attente d'envoi désactivées à l'état inactif (I).</p>
Démarrage d'un abonnement Q	<p>Démarrer une abonnement Q démarre une file d'attente d'envoi inactive si l'abonnement Q que vous démarrez est le seul abonnement Q utilisant la file d'attente d'envoi inactive.</p>
SQL	<p>Utilisez une invite de commande ou un des outils de ligne de commande DB2 pour insérer un signal STARTQ dans la table <code>IBMQREP_SIGNAL</code> sur le serveur Q Capture :</p> <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'STARTQ', 'file_attente_envoi', 'P');</pre> <p>Où <i>schéma</i> identifie un programme Q Capture et <i>file_attente_envoi</i> est le nom de la file d'attente d'envoi que vous voulez démarrer.</p>

Lorsque Q Capture reçoit la commande de démarrage de l'activité de messages sur une ou plusieurs files d'attente d'envoi, il effectue les actions suivantes :

- Changement de l'état des files d'attente pour indiquer un état actif (A) dans la table `IBMQREP_SENDQUEUES`.

- Obtention des informations de redémarrage pour les files d'attente à partir du message de redémarrage.
- Reprise de la lecture du journal au point de redémarrage le plus ancien parmi toutes les files d'attente d'envoi, insertion de messages uniquement dans les files d'attente d'envoi redémarrées jusqu'à ce que toutes les files d'attentes aient le même point de redémarrage.

Arrêt de l'activité de message sur une ou plusieurs files d'attente d'envoi

Vous pouvez indiquer au programme Q Capture d'arrêter de placer des messages dans les files d'attente d'envoi sélectionnées lorsque vous devez arrêter temporaire la réplication ou la publication sur une ou plusieurs files d'attente.

A propos de cette tâche

Par exemple, il peut être nécessaire de corriger une file d'attente désactivée ou d'arrêter l'activité de messages lorsque vous effectuez des tâches d'administration WebSphere MQ.

Lorsque vous arrêtez l'activité de messages sur une ou plusieurs files d'attente d'envoi, le programme Q Capture continue à répliquer ou à publier les modifications pour les abonnements Q actifs ou les publications utilisant d'autres files d'attente d'envoi. L'arrêt de files d'attente sélectionnées est préférable à l'arrêt du programme Q Capture lorsqu'une erreur de file d'attente se produit ou lorsqu'une maintenance est requise.

Vous pouvez utiliser la commande **startq** pour indiquer au programme Q Capture de reprendre l'insertion de messages dans les files d'attente arrêtées. Puisque Q Capture conserve les informations de redémarrage pour chaque file d'attente, il n'est pas nécessaire d'effectuer un nouveau chargement (régénération intégrale) des tables cible lorsque vous utilisez cette méthode.

Procédure

Pour arrêter l'activité de messages sur une file d'attente d'envoi, utilisez l'une des méthodes suivantes :

Méthode	Description
<p>Commande stopq</p>	<p>Vous pouvez utiliser la commande z/OS MODIFY ou la commande asnqccmd pour émettre stopq. La commande vous permet d'arrêter une file d'attente d'envoi unique ou toutes les files d'attente qu'utilise un programme Q Capture.</p> <p>Dans les exemples ci-dessous, la première commande MODIFY indique au programme Q Capture d'arrêter de placer des messages dans la file d'attente d'envoi asn.qm1.dataq. La deuxième commande arrête l'activité de messages sur toutes les files d'attente d'envoi :</p> <pre>F MYQCAP,STOPQ=ASN.QM1.DATAQ</pre> <pre>F MYQCAP,STOPQ=ALL</pre> <p>Les commandes suivantes utilisent asnqccmd pour les mêmes actions :</p> <pre>asnqccmd capture_server=sourcedb capture_schema="asn" stopq="asn.qm1.dataq"</pre> <pre>asnqccmd capture_server=sourcedb capture_schema="asn" stopq=all</pre> <p>Outre une file d'attente individuelle, vous pouvez spécifier le paramètre captureupto, le paramètre stopafter, voire les deux, afin que Q Capture arrête de placer des messages en file d'attente d'une façon contrôlée à partir d'un point indiqué :</p> <p>captureupto Indiquez une valeur d'horodatage complète ou partielle pour ordonner à Q Capture de publier jusqu'à un point particulier, puis d'arrêter de placer des messages en file d'attente. Vous pouvez également indiquer CURRENT_TIMESTAMP ou EOL pour que Q Capture arrête de placer des messages en file d'attente après la fin du journal actif.</p> <p>stopafter Indique que Q Capture arrête de placer des messages en file d'attente lorsque l'une des conditions suivantes est vraie :</p> <p>data_applied Toutes les modifications situées après le point d'arrêt spécifié sont appliquées à la cible.</p> <p>data_sent Tous les messages sont supprimés de la file d'attente de transmission pointant vers le système cible ou le programme Q Apply a traité tous les messages dans le cas d'une file d'attente de réception/d'envoi locale partagée.</p> <p>Par exemple, les commandes MODIFY et asnqccmd ordonnent à Q Capture de publier jusqu'à midi, puis de ne plus placer de messages dans la file d'attente d'envoi ASN.QM1.DATAQ.</p> <pre>F QCAP1,STOPQ=ASN.QM1.DATAQ CAPTUREUPTO=12:00 asnqccmd capture_server=sourcedb capture_schema="asn" stopq="asn.qm1.dataq" captureupto=12:00</pre>

Méthode	Description
Commande stopq (suite)	La syntaxe suivante est également utilisable avec stopq et MODIFY : <pre> F QCAP1,STOPQ=ASN.QM1.DATAQ CAPTUREUPTO=EOL F QCAP1,STOPQ=ASN.QM1.DATAQ CAPTUREUPTO=CURRENT_TIMESTAMP F QCAP1,STOPQ=ASN.QM1.DATAQ STOPAFTER=DATA_APPLIED F QCAP1,STOPQ=ASN.QM1.DATAQ STOPAFTER=DATA_SENT F QCAP1,STOPQ=ASN.QM1.DATAQ CAPTUREUPTO=EOL STOPAFTER=DATA_APPLIED F QCAP1,STOPQ=ASN.QM1.DATAQ CAPTUREUPTO=EOL STOPAFTER=DATA_SENT F QCAP1,STOPQ=ASN.QM1.DATAQ CAPTUREUPTO=CURRENT_TIMESTAMP STOPAFTER=DATA_APPLIED F QCAP1,STOPQ=ASN.QM1.DATAQ CAPTUREUPTO=CURRENT_TIMESTAMP STOPAFTER=DATA_SENT F QCAP1,STOPQ=ASN.QM1.DATAQ CAPTUREUPTO=12:00 STOPAFTER=DATA_APPLIED F QCAP1,STOPQ=ASN.QM1.DATAQ CAPTUREUPTO=12:00 STOPAFTER=DATA_SENT </pre>
SQL	Utilisez une invite de commande ou l'un des outils de ligne de commande DB2 pour insérer un signal STOPQ dans la table <code>IBMQREP_SIGNAL</code> sur le serveur Q Capture : <pre> insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'STOPQ', 'file_attente_envoi', 'P'); </pre> <p>Où <i>schéma</i> identifie un programme Q Capture et <i>file_attente_envoi</i> est le nom de la file d'attente d'envoi que vous voulez arrêter.</p>

Lorsque le programme Q Capture reçoit la commande, il effectue les actions suivantes :

- Arrêt de la publication des modifications pour toutes les publications ou tous les abonnements Q actifs associés aux files d'attente d'envoi arrêtées.
- Poursuite de l'insertion de messages dans les files d'attente d'envoi actives.
- Changement de l'état des files d'attente arrêtées pour indiquer un état inactif (I) dans la table `IBMQREP_SENDQUEUES`.
- Arrêt de la mise à jour des informations de redémarrage dans le message de redémarrage pour toutes les files d'attente d'envoi arrêtées.

Si toutes les files d'attente d'envoi sont arrêtées, le programme Q Capture continue à lire le journal pour rechercher des signaux tels que `CAPSTART`, continue à effectuer des insertions dans ses tables de contrôle et attend les commandes.

Chapitre 16. Exécution d'un programme Q Apply

Un programme Q Apply lit des messages contenant des transactions provenant de tables source et les applique aux cibles définies par des abonnements Q. Vous pouvez utiliser le programme Q Apply à l'aide du centre de réplication, des commandes et des services système. De plus, vous pouvez modifier les paramètres d'exploitation du programme Q Apply de différentes manières.

Démarrage d'un programme Q Apply

Lorsque vous démarrez un programme Q Apply, il commence à lire les messages de transaction des files d'attente et à appliquer les transactions aux tables cible ou aux procédures stockées.

Avant de commencer

- Si vous démarrez le programme Q Apply à partir d'un poste de travail à distance, configurez les connexions au serveur Q Apply.
- Créez et configurez un gestionnaire de files d'attente WebSphere MQ, des files d'attente et d'autres objets nécessaires.
- Assurez-vous que vous disposez des autorisations nécessaires pour les objets de réplication Q et les objets WebSphere MQ.
- Créez les tables de contrôle Q Apply.
- Configurez la base de données ou le sous-système cible pour une utilisation avec le programme Q Apply.
- Si des abonnements Q spécifient un chargement automatique faisant appel à l'utilitaire EXPORT, créez un fichier des mots de passe pour permettre à Q Apply de se connecter au serveur Q Capture.

A propos de cette tâche

Lorsque vous démarrez un programme Q Apply, vous pouvez spécifier des paramètres d'initialisation afin que ce dernier utilise les nouvelles valeurs jusqu'à ce que vous exécutiez l'une des actions suivantes :

- Vous modifiez les valeurs des paramètres pendant l'exécution du programme
- Vous arrêtez et redémarrez le programme, ce qui l'invite à lire la table `IBMQREP_APPLYPARMS` et à utiliser les valeurs enregistrées dans ce dernier.

Procédure

Pour démarrer un programme Q Apply, procédez de l'une des manières suivantes :

Méthode	Description
<p>z/OS</p> <p>Console ou TSO z/OS</p>	<p>Sous z/OS, vous pouvez démarrer un programme Q Apply avec JCL ou comme tâche lancée par le système. Vous pouvez spécifier de nouvelles valeurs de paramètres d'appel lors du démarrage d'un programme Q Apply avec le langage JCL.</p> <p>La longueur totale des paramètres z/OS, que vous pouvez indiquer dans la zone PARM=, est limitée à 100 octets. Pour dépasser cette limite, les programmes de réplication permettent désormais de préciser autant de paramètres que nécessaire dans le fichier SYSIN.</p> <p>Lorsque l'appel JCL comprend l'instruction de définition de données SYSIN, les éléments spécifiés dans le fichier SYSIN sont automatiquement concaténés par le programme Q Apply aux paramètres PARM=. Vous ne pouvez indiquer que des paramètres Q Apply dans le fichier SYSIN. Les paramètres LE doivent être indiqués dans la zone PARM= ou LE _CEE_ENVFILE=DD, suivis d'une barre oblique (/).</p> <p>Exemple :</p> <pre> /* asterisk indicates a comment line // QAPP EXEC PGM=ASNQAPP,PARMS='LE/Q Apply parameters' /* Parameters can be any or no LE parameters and any or /* no Q Apply parameters //SYSIN DD * /* additional Q Apply parameters, one or more /* parameters on each line APPLY_SERVER=DSN!! APPLY_SCHEMA=APPCAT DEBUG=Y LOGSTDOUT=N </pre>
<p>commande asnqapp</p>	<p>Dans une ligne de commande, exécutez la commande asnqapp pour démarrer un programme Q Apply et spécifiez, si vous le souhaitez, des paramètres de démarrage :</p> <pre> asnqapp apply_server=<i>nom_serveur</i> apply_schema=<i>schéma</i> <i>paramètres</i> </pre> <p>Où <i>nom_serveur</i> est le nom de la base de données ou du sous-système où les tables de contrôle Q Apply sont définies et où le programme Q Apply validera les changements dans les cibles, où <i>schéma</i> identifie le programme Q Apply que vous souhaitez démarrer et où <i>paramètres</i> correspond à un ou plusieurs paramètres pouvant être spécifiés lors du démarrage.</p>
<p>Windows</p> <p>Service Windows</p>	<p>Vous pouvez créer un service de réplication dans le système d'exploitation Windows pour démarrer le programme Q Apply automatiquement lors du démarrage du système.</p>

Vous pouvez vérifier si un programme Q Apply a été démarré en procédant de l'une des manières suivantes :

- Dans le fichier journal de diagnostic Q Apply (*serveur_apply.schéma_apply.QAPP.log* sous z/OS et *instancedb2.serveur_apply.schéma_apply.QAPP.log* sous Linux, UNIX, et Windows) recherchez un message indiquant que le programme enregistre les changements.
- Dans la table IBMQREP_APPLYTRACE, recherchez un message indiquant que le programme enregistre les changements.

- **z/OS** Si vous êtes en mode de traitement par lots, examinez la console z/OS ou le journal des travaux z/OS pour voir si des messages indiquent que le programme a démarré.
- Dans la fenêtre Messages Q Apply du Centre de réplication, recherchez un message indiquant que le programme a été démarré. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Apply qui contient le programme Q Apply dont vous voulez visualiser les messages et sélectionnez **Rapports > Messages de Q Apply**.

Changement des paramètres de Q Apply

Vous pouvez modifier les paramètres d'exploitation de Q Apply lors du démarrage du programme, pendant l'exécution de ce dernier ou en mettant à jour la table de contrôle IBMQREP_APPLYPARMS.

Pour une brève description des trois méthodes de changement des paramètres et un exemple vous aidant à comprendre les différences, voir «Méthodes de changement des paramètres d'exploitation de Q Capture», à la page 255. Les méthodes employées sont identiques pour un programme Q Apply.

Modification de paramètres pendant l'exécution d'un programme Q Apply

Vous pouvez modifier le comportement d'un programme Q Apply pendant qu'il poursuit l'application des transactions aux cibles. Le programme Q Apply commence à utiliser les nouveaux paramètres presque immédiatement, mais les changements ne sont pas sauvegardés dans la table de contrôle IBMQREP_APPLYPARMS. Si vous arrêtez puis redémarrez le programme Q Apply, ce dernier utilise les valeurs sauvegardées dans la table de contrôle.

A propos de cette tâche

Vous pouvez modifier les paramètres de Q Apply suivants pendant l'exécution du programme :

- **autostop**
- **logreuse**
- **logstdout**
- **monitor_interval**
- **monitor_limit**
- **prune_interval**
- **term**
- **trace_limit**
- **deadlock_retries**

Procédure

Pour changer dynamiquement des paramètres tandis que le programme Q Apply est en cours d'exécution, utilisez le paramètre **chgparms** avec la commande **MODIFY** sous z/OS ou la commande **asnqacmd** sous Linux, UNIX, Windows et les services système UNIX (USS) sous z/OS

MODIFY

```
f myqapp,chgparms paramètre=valeur
```

Où myqapp correspond au nom du travail Q Apply.

asnrqcmd

```
asnrqcmd apply_server=serveur  
apply_schema=schéma  
chgparms paramètres
```

Où *serveur* correspond au nom du serveur Q Apply, *schéma* représente le programme Q Apply en cours d'exécution, et *paramètres* correspond aux paramètres à modifier.

Un délai de 1 à 2 secondes peut s'écouler entre le moment où vous exécutez la commande et celui où un programme Q Apply change de fonctionnement.

Changement des paramètres enregistrés de Q Apply dans la table IBMQREP_APPLYPARMS

Un programme Q Apply stocke ses paramètres d'exploitation dans la table de contrôle IBMQREP_APPLYPARMS. Si vous écrasez ces paramètres enregistrés lors du démarrage du programme ou pendant son exécution, les modifications sont uniquement conservées dans la mémoire. Lors du prochain démarrage de du programme Q Apply, les valeurs enregistrées dans la table de contrôle seront utilisées. Pour modifier les valeurs des paramètres enregistrées, vous devez mettre la table de contrôle à jour.

A propos de cette tâche

La table IBMQREP_APPLYPARMS contient une seule ligne. Si cette table n'a pas de ligne ou plusieurs lignes, le programme Q Apply ne s'exécutera pas.

Si vous souhaitez modifier une ou plusieurs valeurs de paramètres sauvegardés, vous pouvez mettre à jour chacune des colonnes de la table IBMQREP_APPLYPARMS. Etant donné qu'un programme Q Apply lit cette table lors de son démarrage, il est nécessaire d'arrêter et de redémarrer le programme pour valider les changements.

Procédure

Pour modifier les paramètres enregistrés d'un programme Q Apply dans la table IBMQREP_APPLYPARMS :

Utilisez l'une des méthodes suivantes :

Option	Description
Le tableau de bord de la réplication Q	Sous l'onglet Programmes, cliquez sur Propriétés , puis sélectionnez un serveur Q Apply dans la zone Programme , pour afficher ou modifier les valeurs sauvegardées.

Option	Description
SQL	<p>A partir d'une invite de commande ou de l'un des outils de ligne de commande DB2, exécutez l'instruction SQL UPDATE pour la table IBMQREP_APPLYPARMS. Par exemple, pour modifier les valeurs par défaut des paramètres prune_interval et deadlock_retries :</p> <pre>update nom_schéma.ibmqrep_applyparms set prune_interval=600, deadlock_retries=10</pre> <p>Où <i>schéma</i> identifie le programme Q Apply dont vous souhaitez modifier les valeurs des paramètres enregistrés.</p>
Centre de répliation	<p>Utilisez la fenêtre Changer les paramètres - Sauvegardés pour visualiser ou modifier l'une des valeurs de la table IBMQREP_APPLYPARMS. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Apply qui contient le programme Q Apply dont vous voulez visualiser ou modifier les paramètres sauvegardés et sélectionnez Changer les paramètres > Sauvegardés.</p> <p>Remarque : En démarrant avec la version 10.1, les nouveaux paramètres Q Apply n'ont pas tous été ajoutés au Centre de répliation. Pour voir tous les paramètres, utilisez le tableau de bord de répliation Q.</p>

Arrêt d'un programme Q Apply

Vous pouvez arrêter un programme Q Apply via le Centre de répliation ou des commandes système.

A propos de cette tâche

Lors de l'arrêt d'un programme Q Apply, les actions suivantes sont effectuées :

- Arrêt de la lecture des messages de toutes les files d'attente de réception
- Annulation des transactions ayant été partiellement appliquées aux cibles, mais pas validées
- Arrêt de manière ordonnée

Lors de l'arrêt d'un programme Q Apply, les messages des programmes Q Capture en cours d'exécution continuent d'être collectés dans les files d'attente de réception. Lorsque vous redémarrez le programme Q Apply, ce dernier commence la lecture de ces messages et lit de nouveau tous les messages contenant des transactions annulées. Le programme reprend alors l'application des transactions aux cibles. Les transactions ne sont appliquées qu'une seule fois et aucune donnée répliquée n'est perdue.

Avertissement :

- Si vous arrêtez le programme Q Apply pendant le chargement d'une table cible, assurez-vous qu'aucune application ne puisse mettre à jour la table cible avant le redémarrage du programme Q Apply et le chargement de la table. Lors du redémarrage du programme, le contenu de la table cible est supprimé et le chargement de la table reprend. Toute mise à jour des tables cible se produisant pendant l'arrêt du programme Q Apply est perdue.
- Durant le processus de chargement, le programme Q Apply supprime les contraintes d'intégrité référentielle dans les tables cible. Ces contraintes ne sont pas réappliquées jusqu'au redémarrage du programme et au chargement de la table. L'intégrité référentielle des mises à jour effectuées dans les tables cible pendant l'arrêt du programme Q Apply ne sera pas vérifiée.

Si vous souhaitez arrêter la lecture des messages d'une seule file d'attente de réception par un programme Q Apply, exécutez la commande **asnqacmd stopq**.

Procédure

Pour arrêter un programme Q Apply, procédez de l'une des manières suivantes :

Méthode	Description
Paramètre stop	Utilisez ce paramètre avec la commande MODIFY sous z/OS ou la commande asnqacmd sous Linux, UNIX, Windows et UNIX System Services sous z/OS pour arrêter un programme Q Apply : MODIFY f myqapp,stop Où myqapp correspond au nom du travail Q Apply. asnqacmd asnqacmd apply_server= <i>nom_serveur</i> apply_schema= <i>schéma</i> stop Où <i>nom_serveur</i> est le nom de la base de données ou du sous-système où le programme Q Apply est en cours d'exécution et <i>schéma</i> identifie le programme Q Apply que vous souhaitez arrêter.
Windows Service Windows	Vous pouvez créer un service de réplication DB2 dans le système d'exploitation Windows, et utiliser le gestionnaire de contrôle de service Windows ou la commande net stop pour arrêter un programme Q Apply.

Arrêt du traitement des messages dans une file d'attente de réception

Vous pouvez utiliser le Centre de réplication ou le paramètre **stopq** pour demander au programme Q Apply de cesser le traitement des messages contenus dans une file d'attente de réception.

Avant de commencer

- Le programme Q Apply doit être en cours d'exécution.
- La file d'attente de réception doit être à l'état A (actif).

A propos de cette tâche

Lors de l'arrêt du traitement pour une file d'attente :

- Le programme Q Apply continue d'appliquer les transactions des autres files d'attente de réception.
- Si des abonnements Q utilisant la file d'attente sont actifs, les messages continuent d'arriver dans la file d'attente de réception pour laquelle le traitement a cessé.

Vous pouvez reprendre le traitement des messages dans la file d'attente de réception et aucun message ne sera perdu.

Si vous arrêtez le traitement des messages d'une file d'attente de réception, le programme Q Apply modifie l'état de la file d'attente à I (inactif) dans la table IBMQREP_RECVQUEUES. Même si le programme Q Apply est arrêté puis redémarré, il ne reprend la lecture dans la file d'attente que si vous exécutez la commande **asqcmd startq**.

Lors de l'arrêt du traitement des messages dans une file d'attente de réception, le programme Q Apply termine l'application des transactions stockées de la file d'attente. Le programme Q Apply annule les transactions partiellement appliquées dans les cibles. Les transactions annulées seront traitées lors du prochain démarrage du programme Q Apply. Aucune donnée n'est perdue.

Procédure

Pour cesser le traitement des messages contenus dans une file d'attente de réception :

Utilisez l'une des méthodes suivantes :

Méthode	Description
Paramètre stopq	<p>Utilisez ce paramètre avec la commande MODIFY sous z/OS ou la commande asqcmd sous Linux, UNIX, Windows et UNIX System Services sous z/OS pour arrêter le traitement des messages sur une file d'attente de réception :</p> <p>MODIFY f myqapp,stopq</p> <p>Où myqapp correspond au nom du travail Q Apply.</p> <p>asqcmd asqcmd apply_server=<i>nom_serveur</i> apply_schema=<i>schéma</i> stopq</p> <p>Où <i>nom_serveur</i> est le nom de la base de données ou du sous-système où le programme Q Apply est en cours d'exécution et <i>schéma</i> identifie le programme Q Apply pour lequel vous souhaitez arrêter le traitement des messages.</p>
Centre de répliation	<p>Utilisez la fenêtre Gérer les files d'attente de réception. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Apply où la file d'attente de réception se trouve et sélectionnez Gérer > Files d'attente de réception.</p>

Pour s'assurer que le programme Q Apply a cessé la lecture des messages de la file d'attente :

- Vérifiez si l'état de la file d'attente est devenu I (inactif) dans la fenêtre Gérer des files d'attente de réception.

- Dans le fichier journal de diagnostic de Q Apply, recherchez un message indiquant l'arrêt du traitement pour la file d'attente en question.
- Dans la table de contrôle IBMQREP_RECVQUEUEES, vérifiez que l'état de la file d'attente est passé à I (inactif) dans la colonne STATE :

```
SELECT RECVQ, STATE FROM schéma.IBMQREP_RECVQUEUEES
OÙ RECVQ = 'nom_file_réception';
```

Démarrage du traitement de messages dans une file d'attente de réception

Vous pouvez utiliser le Centre de réplication ou une commande pour indiquer au programme Q Apply qu'il doit commencer le traitement des messages contenus dans une file d'attente de réception.

Avant de commencer

- Le programme Q Apply doit être en cours d'exécution.
- La file d'attente de réception doit être à l'état I (inactif).

A propos de cette tâche

Vous devrez peut-être demander à un programme Q Apply de commencer le traitement des messages dans une file d'attente de réception pour plusieurs motifs :

- Un conflit, une erreur SQL ou des blocages persistants survenus au niveau de la cible, ayant entraîné l'arrêt du programme Q Apply ou l'arrêt de la lecture dans la file d'attente de réception.
- Vous avez demandé au programme Q Apply de cesser le traitement des messages dans la file d'attente de réception.

Remarque : Vous pouvez utiliser le Centre de réplication ou le paramètre d'invocation **startallq** pour indiquer au programme Q Apply qu'il doit commencer la lecture dans toutes les files d'attente de réception lorsque le programme démarre, même si les files d'attente sont dans l'état inactif (I)

Procédure

Pour commencer le traitement des messages contenus dans une file d'attente de réception :

Utilisez l'une des méthodes suivantes :

Méthode	Description
Paramètre startq	<p>Utilisez ce paramètre avec la commande MODIFY sous z/OS ou la commande asnqcmd sous Linux, UNIX, Windows et UNIX System Services sous z/OS pour démarrer le traitement des messages sur une file d'attente de réception :</p> <p>MODIFY f myqapp, startq</p> <p>Où myqapp correspond au nom du travail Q Apply.</p> <p>asnqcmd asnqcmd apply_server=<i>nom_serveur</i> apply_schema=<i>schéma</i> startq</p> <p>Où <i>nom_serveur</i> est le nom de la base de données ou du sous-système où le programme Q Apply est en cours d'exécution et <i>schéma</i> identifie le programme Q Apply pour lequel vous souhaitez démarrer le traitement des messages.</p>
Centre de réplication	<p>Utilisez la fenêtre Gérer les files d'attente de réception. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Apply où la file d'attente de réception se trouve et sélectionnez Gérer > Files d'attente de réception.</p>

Pour s'assurer que le programme Q Apply a démarré la lecture des messages dans la file d'attente :

- Vérifiez si l'état de la file d'attente est passé à A (actif) dans la fenêtre Gérer les files d'attente de réception.
- Dans le fichier journal de diagnostic de Q Apply, recherchez un message indiquant le début du traitement pour la file d'attente en question.
- Dans la table de contrôle IBMQREP_RECVQUEUES, vérifiez que l'état de la file d'attente est passé à A (actif) dans la colonne STATE :

```
SELECT RECVQ, STATE FROM schéma.IBMQREP_RECVQUEUES
OÙ RECVQ = 'nom_file_reception';
```

Indication des transactions à ignorer par un programme Q Apply

Vous pouvez demander à un programme Q Apply d'ignorer des transactions qui seront retirées de la file d'attente de réception mais non appliquées aux tables cible.

A propos de cette tâche

Pour ignorer une ou plusieurs transactions, vous pouvez utiliser la commande **asnqapp** lorsque vous démarrez le programme Q Apply. Vous pouvez également spécifier d'ignorer une ou plusieurs transactions sur une file d'attente de réception lorsque vous démarrez la file d'attente de réception à l'aide de la commande **startq**.

Il peut être utile d'empêcher le traitement de certaines transactions par le programme dans certaines situations imprévues, comme par exemple :

- Le programme Q Apply reçoit une erreur lors de l'application d'une ligne d'une transaction, et s'arrête complètement ou arrête de lire la file d'attente de réception. Au démarrage, vous souhaitez peut-être que le programme Q Apply ignore l'ensemble de la transaction qui a provoqué l'erreur.

- Après un basculement dans le cadre d'une reprise après incident, vous préférerez peut-être ignorer une plage de transactions de la file d'attente de réception, depuis le noeud défaillant vers le noeud de secours.
- Un grand nombre d'opérations DELETE sont répliquées, ralentissant le traitement du programme Q Apply. Si vous n'avez pas besoin de supprimer les lignes de la table cible, vous pouvez ignorer les transactions contenant les lignes supprimées et ainsi améliorer les performances globales.

Vous pouvez également demander au programme Q Capture d'ignorer des transactions. Cette action est assez classique dans les cas où vous pouvez planifier les transactions qui n'ont pas besoin d'être répliquées.

Remarque : Toutefois, si vous ignorez une transaction qui avait été validée sur le serveur source, vous risquez d'avoir des divergences entre les tables source et cible. Il se peut que vous deviez exécuter d'autres actions pour synchroniser les tables.

Restrictions

La capacité à ignorer les transactions sur la cible n'est pas prise en charge sur les sources de réplification Classic.

Procédure

Pour demander à un programme Q Apply d'ignorer des transactions, utilisez l'une des méthodes suivantes :

Situation	Méthode
<p>Au démarrage du programme</p>	<p>Utilisez la commande asnqapp avec le paramètre skiptrans pour demander au programme Q Apply de ne pas appliquer une ou plusieurs transactions d'une ou plusieurs files d'attente de réception lors de son initialisation. Le format de la commande est le suivant :</p> <pre>asnqapp apply_server=serveur apply_schema=schéma skiptrans=nom de file d'attente de réception;transaction_ID</pre> <p>Vous pouvez trouver <i>ID_transaction</i> dans l'entrée qTransMsgHeader.uow_id de la sortie de la commande asnqfmt pour une file d'attente de réception ou dans la colonne SRC_TRANS_ID de la table IBMQREP_EXCEPTIONS si une transaction en erreur doit être ignorée. L'ID se présente sous la forme d'un identifiant hexadécimal de 10 octets, au format suivant :</p> <p>z/OS</p> <pre>0000:xxxx:xxxx:xxxx:mmmm</pre> <p>Où <i>xxxx:xxxx:xxxx</i> correspond à l'ID de transaction et <i>mmmm</i> à l'ID membre pour le partage de données. Vous trouverez l'ID membre dans les deux derniers octets de l'en-tête de l'enregistrement de journal, dans la sortie LOGP. L'ID membre est 0000 si le partage de données n'est pas activé.</p> <p>Par exemple, le JCL suivant spécifie qu'une transaction doit être ignorée au sein d'un environnement de partage des données, avec un ID de membre 0001 :</p> <pre>//QAPP EXEC PGM=ASNQAPP, //PARM='/APPLY_SERVER=DSN1 APPLY_SCHEMA=APPCAT //SKIPTRANS=Q1;0000:1171:12c3:0890:0001</pre> <p>Si la limite de ligne de commande autorisée est dépassé par la syntaxe ci-dessus, vous pouvez également utiliser le fichier SYSIN pour spécifier les transactions à ignorer. L'instruction de définition de données de l'exemple suivant pour le travail Q Apply spécifie d'ignorer une transaction sur la file d'attente Q1 et une plage de transactions sur la file d'attente Q3 :</p> <pre>//SYSIN DD * SKIPTRANS=Q1;0000:1171:12c3:0890:000, Q3;0000:0000:0000:79bc:0000-0000:0000:0000:79fc:0000</pre> <p>Linux UNIX Windows</p> <pre>mmmm:0000:xxxx:xxxx:xxxx</pre> <p>Où <i>xxxx:xxxx:xxxx</i> représente l'ID de transaction et <i>mmmm</i>, l'identificateur de partition pour les bases de données partitionnées (0000 pour les bases de données non partitionnées).</p> <p>Par exemple, la commande suivante spécifie qu'une transaction doit être ignorée au sein d'une base de données non partitionnée :</p> <pre>asnqapp capture_server=target apply_schema=ASN skiptrans="recvq1;0000:0000:0000:0000:BE97"</pre> <p>Vous pouvez spécifier d'ignorer une plage de transactions en séparant les ID de transaction par un trait d'union, par exemple :</p> <pre>skiptrans= "Q1;0000:0000:0000:0000:51b0-0000:0000:0000:0000:51g0"</pre>

Situation	Méthode
<p>Lors du démarrage d'une file d'attente de réception</p>	<p>Lorsque vous démarrez une file d'attente de réception avec le paramètre startq de la commande asnacmd ou MODIFY, spécifiez le paramètre skiptrans pour ignorer une transaction ou une plage de transactions de la file d'attente de réception que vous démarrez. Le format de la commande pour asnacmd est le suivant :</p> <pre>asnacmd apply_server=serveur apply_schema=schéma startq=nom_file_d'attente;skiptrans="ID_transaction"</pre> <p>Par exemple, la commande suivante demande au programme Q Apply d'ignorer une transaction sur la file d'attente de réception Q1 :</p> <pre>asnacmd apply_server=target apply_schema=ASN startq=Q1;skiptrans="0000:0000:0000:0000:51a1"</pre>

Lorsqu'une ou plusieurs transactions sont ignorées avec succès, le programme Q Apply émet des messages ASN7670I ou ASN7671I dans son journal et dans la table IBMQREP_APPLYTRACE.

Remarque concernant le chargement de la table cible : Si vous démarrez le programme Q Apply avec le paramètre **skiptrans**, les transactions qui font partie d'un chargement de table cible automatique (effectué par Q Apply) ne sont pas appliquées. Si vous avez spécifié un chargement manuel, ces transactions peuvent être appliquées aux tables cible dans le cadre d'une opération de chargement qui est externe à la file d'attente de réception. Il est possible que vous soyez obligé de supprimer manuellement ces lignes à partir de la cible.

Chapitre 17. Modification de l'environnement de réplication Q

Vous pouvez modifier les propriétés des objets de réplication tels que les abonnements Q et les mappes de file d'attente de réplication après les avoir définies, dans la plupart du cas, sans interrompre la réplication.

Modification des propriétés des abonnements Q unidirectionnels

Vous pouvez changer les propriétés des abonnements Q sans arrêter la réplication. Dans un premier temps, vous faites vos changements. Vous réinitialisez ensuite un seul abonnement Q ou le programme Q Capture si vous changez plusieurs abonnements Q.

Restrictions

Vous ne pouvez pas modifier toutes les propriétés des abonnements Q.

- Si vous utilisez le programme de ligne de commande ASNCLP pour effectuer les modifications, vérifiez la syntaxe de la commande ALTER QSUB pour votre réplication unidirectionnelle afin de voir les propriétés que vous pouvez modifier.
- Dans le Centre de réplication, les contrôles sont désactivés dans le bloc-notes Propriétés des abonnements Q pour les propriétés que vous ne pouvez pas modifier.

A propos de cette tâche

Si vous réinitialisez un programme Q Capture, cette action va aussi réinitialiser toute publication définie dans le schéma de Q Capture. Une telle réinitialisation requiert également davantage de ressources système.

Si vous n'avez besoin de modifier que les propriétés d'un ou deux abonnements Q unidirectionnels, il est moins onéreux, en termes de ressources, de réinitialiser un abonnement Q à la fois.

Procédure

Pour modifier les propriétés des abonnements Q unidirectionnels sans arrêter la réplication :

1. Modifiez un ou plusieurs abonnements Q. Utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande ALTER QSUB. Par exemple, les commandes suivantes définissent l'environnement et modifient l'abonnement Q unidirectionnel pour l'abonnement Q EMPLOYEE0001 sur le serveur exemple : <pre>ASNCLP SESSION SET TO Q REPLICATION; SET RUN SCRIPT NOW STOP ON SQL ERROR ON; SET SERVER CAPTURE TO SAMPLE; SET SERVER TARGET TO TARGET; SET CAPTURE SCHEMA SOURCE ASN1; SET APPLY SCHEMA ASN1; ALTER QSUB EMPLOYEE0001 REPLQMAP SAMPLE_ASN1_TO_TARGETDB_ASN1 USING OPTIONS ALL CHANGED ROWS N HAS LOAD PHASE I SUPPRESS DELETES N CONFLICT ACTION F;</pre>
Centre de réplication	Utilisez le bloc-notes Propriétés des abonnements Q. Pour ouvrir le bloc-notes, cliquez avec le bouton droit de la souris sur un abonnement Q puis sélectionnez Propriétés .

2. Si vous changez un seul abonnement Q, réinitialisez-le. Utilisez l'une des méthodes suivantes :

Méthode	Description
Centre de réplication	Utilisez la fenêtre Gérer les abonnements Q. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture où la table source de l'abonnement Q est localisée puis sélectionnez Gérer > Abonnements Q .
SQL	Insérez un signal REINIT_SUB dans la table IBMQREP_SIGNAL au niveau du serveur Q Capture : <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'REINIT_SUB', 'sous-nom', 'P');</pre> <p>où <i>schéma</i> identifie le programme Q Capture en train de traiter l'abonnement Q et <i>sous-nom</i> est le nom de l'abonnement Q que vous souhaitez réinitialiser.</p>

3. Si vous changez plusieurs abonnements Q, réinitialisez le programme Q Capture. Utilisez l'une des méthodes suivantes :

Méthode	Description
Paramètre reinit	<p>Associez ce paramètre à la commande MODIFY sous z/OS ou à la commande asnqccmd sous Linux, UNIX, Windows et les services système UNIX (USS) sous z/OS pour réinitialiser tous les abonnements Q d'un schéma Q Capture :</p> <p>MODIFY f myqcap, reinit</p> <p>Où myqcap est le nom du travail Q Capture.</p> <p>asnqccmd asnqccmd capture_server=<i>nom_serveur</i> capture_schema=<i>schéma</i> reinit</p> <p>où <i>nom_serveur</i> est le nom de la base de données ou du sous-système sur lequel le programme Q capture fonctionne et <i>schéma</i> identifie le programme Q Capture à partir duquel vous souhaitez réinitialiser tous les abonnements Q.</p>
Centre de réplication	<p>Utilisez la fenêtre Réinitialiser Q Capture. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture contenant le programme Q Capture que vous souhaitez réinitialiser puis sélectionnez Réinitialiser le programme Q Capture.</p>

Ajout de colonnes existantes à un abonnement Q (réplication unidirectionnelle)

Vous pouvez ajouter des colonnes existantes à un abonnement Q unidirectionnel à partir de la table source lors de l'exécution des programmes de réplication. Si les colonnes n'existent pas et que vos serveurs sont à la version 10.1 ou ultérieure, utilisez une procédure différente.

Avant de commencer


- Les abonnements Q auxquels les colonnes sont ajoutées doivent avoir pour état A (active).
- **Linux UNIX Windows** Si le type de données de la colonne est LONG VARCHAR ou GRAPHIC, la base de données source ou le sous-système doit être configuré avec DATA CAPTURE CHANGES INCLUDE VARCHAR COLUMNS.

Restrictions

- Les colonnes que vous ajoutez doivent avoir pour valeur NULL admis ou être définies comme NOT NULL WITH DEFAULT.
- Si vous ajoutez des colonnes avec des valeurs par défaut, vous devez exécuter l'utilitaire REORG dans la table source avant de démarrer la réplication de la nouvelle colonne. Pour plus d'informations, voir Eviter les erreurs CHECKFAILED lors de l'ajout de colonnes aux tables cible DB2 for z/OS.
- Vous ne pouvez pas ajouter plus de 20 colonnes au cours d'un intervalle de validation Q Capture, tel que spécifié par le paramètre **commit_interval**.
- **Cibles fédérées** : pour ajouter des colonnes à un abonnement Q existant, vous pouvez utiliser le signal ADDCOL mais vous devez déplacer l'abonnement puis le créer de nouveau après avoir modifié la table cible car vous ne pouvez pas ajouter de colonnes à un pseudonyme.

A propos de cette tâche

Pour la version 10.1 ou ultérieure : Vous ne devez pas utiliser cette procédure lors de l'ajout de nouvelles colonnes à la table source si les serveurs participants sont à la version 10.1 ou ultérieure sous z/OS et Linux, UNIX et Windows. Si vous définissez la valeur de la colonne REPL_ADDCOL à Y dans la table IBMQREP_SUBS, lorsque vous ajoutez de nouvelles colonnes à une table, les colonnes sont ajoutées automatiquement à l'abonnement Q, puis à la table cible si elles n'existent pas déjà dans celle-ci. Pour effectuer ce réglage, indiquez REPLICATE ADD COLUMN YES dans la commande ASNCLP CREATE QSUB ou activez la case à cocher **Automatically replicate new columns added to the source table** (Répliquer automatiquement de nouvelles colonnes ajoutées à la table source) lorsque vous créez ou modifiez les propriétés d'un abonnement Q dans le Centre de réplication. Pour plus d'informations, consultez ces rubriques :

-  Enabling replication of ADD COLUMN and SET DATA TYPE operations (Activation de la réplication des opérations ADD COLUMN et SET DATA TYPE)
- «Activation d'une réplication automatique des colonnes nouvellement ajoutées à partir de la table source», à la page 140

Lorsque vous insérez le signal dans le serveur Q Capture, la colonne est ajoutée automatiquement à la table cible si vous ne l'avez pas déjà fait. Si vous souhaitez ajouter plusieurs colonnes à un abonnement Q, insérez un signal pour chaque nouvelle colonne. Vous pouvez ajouter plusieurs colonnes dans une seule transaction. Il est possible d'arrêter Q Capture pendant que vous insérez les signaux qu'il lira depuis le journal lorsqu'il redémarrera.

Laissez les programmes de réplication ajouter automatiquement les nouvelles colonnes à la table source pour garantir qu'elles correspondent aux colonnes de la table source. Les colonnes sont ajoutées à la table cible avec le même type de données, la même caractéristique null et la même valeur par défaut que les colonnes correspondantes de la table source. Vous pouvez spécifier un autre nom de colonne cible si vous utilisez la commande ALTER ADD COLUMN du programme de ligne de commande ASNCLP ou un signal ADDCOL.

Procédure

Pour ajouter des colonnes à un abonnement unidirectionnel, appliquez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande ALTER ADD COLUMN. Par exemple, la commande suivante ajoute la colonne BONUS à l'abonnement DEPARTMENT0001 : <pre>ALTER ADD COLUMN USING SIGNAL (BONUS) QSUB DEPARTMENT0001 USING REPMAP SAMPLE_ASN_TO_TARGET_ASN;</pre>
Tableau de bord de la réplication Q	Dans l'onglet Abonnements, sélectionnez un abonnement Q dans la table et cliquez sur Actions > Ajouter des colonnes .

Méthode	Description
SQL	<p>Utilisez une invite de commande ou l'un des outils de ligne de commande DB2 pour insérer un signal ADDCOL dans la table IBMQREP_SIGNAL au niveau du serveur Q Capture. Par exemple :</p> <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'ADDCOL', 'sous-nom;nom_colonne;nom_colonne_précédente; nom_colonne_suivante', 'P');</pre> <p><i>schéma</i> Identifie le programme Q Capture qui traite l'abonnement Q auquel vous êtes en train d'ajouter une colonne.</p> <p><i>sous-</i> <i>nom;nom_colonne;nom_colonne_précédente;nom_colonne_suivante</i> Nom de l'abonnement Q auquel vous voulez ajouter une colonne et nom de la colonne que vous ajoutez, séparés par un point-virgule. Ces noms sont sensibles à la casse. Il n'est donc pas nécessaire d'utiliser les guillemets pour préserver la casse. Suivez ces exemples :</p> <p>Ajout de colonne dans la table source pour l'abonnement Q et la table cible QSUB1;COL10</p> <p>Ajout de colonne et d'image-avant de la colonne (pour des tables cibles CCD) QSUB1;COL10;XCOL10</p> <p>Ajout de colonne sans image-avant mais avec un nom de colonne cible différent QSUB1;COL10;;TRGCOL10 (placez deux points-virgules (;)) pour indiquer que vous omettez la colonne d'image-avant.)</p>

Après avoir traité le signal, Q Capture commence à enregistrer les changements de la nouvelle colonne lorsqu'il lit les données de journal qui englobent la colonne. Les changements de colonnes qui sont validés après la validation de l'insertion du signal ADDCOL seront répliqués vers la nouvelle colonne dans la table cible. Les lignes qui existaient avant l'ajout de la nouvelle colonne auront une valeur null ou une valeur par défaut pour la nouvelle colonne.

Ajout de colonnes existantes à un abonnement Q (réplication bidirectionnelle ou entre homologues)

Vous pouvez ajouter des colonnes existantes à un abonnement Q bidirectionnel ou entre homologues à partir de la table source lors de l'exécution des programmes de réplication. Si les colonnes n'existent pas et que vos serveurs sont à la version 10.1 ou ultérieure, utilisez une procédure différente.

Avant de commencer

- Les abonnements Q indiquant la table doivent être à l'état A (actif) sur tous les serveurs.
- **Linux UNIX Windows** Si le type de données de la colonne est LONG VARCHAR ou GRAPHIC, la base de données source ou le sous-système doit être configuré avec DATA CAPTURE CHANGES INCLUDE VARCHAR COLUMNS.

Restrictions

- Les colonnes que vous ajoutez doivent avoir pour valeur NULL admis ou être définies comme NOT NULL WITH DEFAULT.
- Si vous ajoutez des colonnes avec des valeurs par défaut, vous devez exécuter l'utilitaire REORG dans la table source avant de démarrer la réplication de la nouvelle colonne. Pour plus d'informations, voir Eviter les erreurs CHECKFAILED lors de l'ajout de colonnes aux tables cible DB2 for z/OS.
- Vous ne pouvez pas modifier la valeur par défaut d'une colonne que vous venez d'ajouter tant que le signal ADDCOL pour cette colonne n'est pas traité.
- Vous ne pouvez pas ajouter plus de 20 colonnes au cours d'un intervalle de validation Q Capture, tel que spécifié par le paramètre **commit_interval**.

A propos de cette tâche

Pour la version 10.1 ou ultérieure : Vous ne devez pas utiliser cette procédure lors de l'ajout de nouvelles colonnes à la table source si les serveurs participants sont à la version 10.1 ou ultérieure sous z/OS et Linux, UNIX et Windows. Si vous définissez la valeur de la colonne REPL_ADDCOL à Y dans la table IBMQREP_SUBS, lorsque vous ajoutez de nouvelles colonnes à une table, les colonnes sont ajoutées automatiquement à l'abonnement Q, puis à la table cible si elles n'existent pas déjà dans celle-ci. Pour effectuer ce réglage, indiquez REPLICATE ADD COLUMN YES dans la commande ASNCLP CREATE QSUB ou activez la case à cocher **Automatically replicate new columns added to the source table** (Répliquer automatiquement de nouvelles colonnes ajoutées à la table source) lorsque vous créez ou modifiez les propriétés d'un abonnement Q dans le Centre de réplication. Pour plus d'informations, consultez ces rubriques :

- **z/OS** Enabling replication of ADD COLUMN and SET DATA TYPE operations (Activation de la réplication des opérations ADD COLUMN et SET DATA TYPE)
- «Activation d'une réplication automatique des colonnes nouvellement ajoutées à partir de la table source», à la page 140

Pour utiliser cette procédure, vous devez tout d'abord modifier une table d'un serveur pour pouvoir ajouter une colonne. Vous insérez ensuite un signal SQL dans le serveur. Lorsque le signal est traité, les versions de la table sur les autres serveurs sont automatiquement modifiées pour ajouter la colonne, sauf si vous l'avez ajoutée manuellement. Ce signal ajoute également la colonne aux définitions des abonnements Q sur tous les serveurs.

Vous pouvez ajouter tout nombre de colonnes dans la table source, à tout moment. Vous pouvez faire cela lorsque les programmes Q Capture et Q Apply sont en cours d'exécution ou sont arrêtés.

Recommandation : Insérez un signal ADDCOL à la fois et émettez une commande COMMIT avant d'insérer un nouveau signal ADDCOL ou d'effectuer toute autre transaction.

Procédure

Ajouter des colonnes à répliquer dans une réplification bidirectionnelle ou dans une réplification entre homologues :

1. Modifiez la table logique sur l'un des serveurs pour ajouter la colonne.
Si l'opération ALTER TABLE qui ajoute la colonne à la table source échoue, tous les abonnements Q dans le groupe de réplification entre homologues seront désactivés.
2. Utilisez l'une des méthodes suivantes pour signaler à Q Capture que vous souhaitez ajouter la colonne à l'abonnement Q pour la table source.

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande ALTER ADD COLUMN. Par exemple, la commande suivante ajoute la colonne BONUS à l'abonnement DEPARTMENT0001 : ALTER ADD COLUMN USING SIGNAL (BONUS) QSUB DEPARTMENT0001 USING REPMAP SAMPLE_ASN_TO_TARGET_ASN;
Tableau de bord de réplification Q	Dans l'onglet Abonnements, sélectionnez un abonnement Q dans la table et cliquez sur Actions > Ajouter des colonnes .
SQL	Utilisez une invite de commande ou l'un des outils de ligne de commande DB2 pour insérer un signal ADDCOL dans la table IBMQREP_SIGNAL au niveau du serveur Q Capture. Par exemple : insert into <i>schéma</i> .IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'ADDCOL', ' <i>sous-nom;nom_colonne</i> ', 'P'); <i>schéma</i> Identifie Q Capture sur le serveur où vous avez modifié la table. <i>sous-nom;nom_colonne</i> Nom de l'abonnement Q qui part du serveur Q capture où vous avez modifié la table et nom de la colonne que vous ajoutez, séparés par un point-virgule. Ces noms sont sensibles à la casse. Il n'est donc pas nécessaire d'utiliser les guillemets pour préserver la casse.

Soit l'exemple suivant : une configuration entre homologues a trois serveurs, ServerA, ServerB et ServerC, et six abonnements Q, subA2B, subB2A, subA2C, subC2A, subB2C et subC2B pour la table EMPLOYEE.

Vous ajoutez une colonne ADDRESS à la table EMPLOYEE sur le serveur ServerA. Vous insérez ensuite un signal ADDCOL pour l'abonnement Q qui traite les transactions du serveur ServerA vers le serveur ServerB puis spécifiez subA2B;ADDRESS comme nom de l'abonnement Q et nom de colonne. Un signal ADDCOL seulement est requis. Les programmes de réplication ajoutent automatiquement la colonne ADDRESS dans les tables EMPLOYEE des serveurs ServerB et ServerC puis la définition de colonne aux six abonnements Q.

Gestion des opérations DDL par Q Capture au niveau de la base de données source

Le programme Q Capture réplique automatiquement certaines opérations DDL au niveau de la base de données source. D'autres changements DDL nécessitent une action de votre part sur la base de données cible.

Le tableau suivant décrit comment Q Capture gère différents types d'opérations DDL et comment agir sur tout abonnement Q affecté.

Tableau 25. Gestion des changements DDL à la base de données source par Q Capture et mode d'action

Opération de langage de définition de données	Comment elle est traitée	Action à accomplir
CREATE TABLE	<p>Version 10.1 sous Linux, UNIX et Windows Répliqué automatiquement si la nouvelle table source correspond au modèle d'attribution des noms des schémas et des tables dans l'abonnement au niveau schéma. Lorsque le programme Q Capture détecte dans le journal une opération CREATE TABLE qui correspond à un abonnement au niveau schéma, il informe le programme Q Apply afin de créer une table cible correspondante. Un abonnement Q au niveau table est également créé et mappe les nouvelles tables source et cible.</p> <p>z/OS ou versions antérieures sous Linux, UNIX et Windows Aucune réplication automatique des opérations CREATE TABLE.</p>	<p>Version 10.1 sous Linux, UNIX et Windows Vérifiez que la table source nouvellement créée correspond au modèle d'attribution des noms des schémas et des tables dans un abonnement au niveau schéma.</p> <p>z/OS ou versions antérieures sous Linux, UNIX et Windows Créez un abonnement Q de niveau table pour la nouvelle table source et utilisez les outils d'administration de réplication pour créer une table cible correspondante ou utiliser une table cible existante.</p>

Tableau 25. Gestion des changements DDL à la base de données source par Q Capture et mode d'action (suite)

Opération de langage de définition de données	Comment elle est traitée	Action à accomplir
DROP TABLE	<p>Version 10.1 sous Linux, UNIX et Windows Répliqué automatiquement si la table source fait partie d'un souscription de niveau schéma. Lorsque le programme Q Capture détecte dans le journal une opération DROP TABLE qui correspond à un abonnement de niveau schéma, les abonnements Q de niveau table associés pour toutes les files d'attente sont également supprimés.</p> <p>z/OS ou versions antérieures sous Linux, UNIX et Windows Q Capture laisse l'abonnement Q actif mais la table source n'a aucun enregistrement de journal à lire. Sous z/OS, le message d'avertissement ASN0197W est émis.</p>	<p>Version 10.1 sous Linux, UNIX et Windows Vérifiez que la table source est incluse dans un abonnement de niveau schéma.</p> <p>z/OS ou versions antérieures sous Linux, UNIX et Windows Lorsque vous supprimez une table, l'abonnement Q pour la table existe toujours. Pour la suppression, arrêtez l'abonnement Q, puis supprimez-le.</p>

Tableau 25. Gestion des changements DDL à la base de données source par Q Capture et mode d'action (suite)

Opération de langage de définition de données	Comment elle est traitée	Action à accomplir
<p>ALTER TABLE ADD (COLUMN)</p>	<p>Version 10.1 sous z/OS et Linux, UNIX ou Windows ou ultérieure Si vous définissez la valeur de la colonne REPL_ADDCOL à Y dans la table IBMQREP_SUBS, lorsque vous ajoutez de nouvelles colonnes à une table, les colonnes sont ajoutées automatiquement à l'abonnement Q, puis à la table cible si elles n'existent pas déjà dans celle-ci.</p> <p>Versions antérieures Q Capture laisse l'abonnement Q actif mais ne réplique pas la colonne ajoutée jusqu'à la réception d'un signal ADDCOL.</p>	<p>Version 10.1 sous z/OS et Linux, UNIX ou Windows ou ultérieure Indiquez REPLICATE ADD COLUMN YES dans la commande ASNCLP CREATE QSUB ou activez la case à cocher Automatically replicate new columns added to the source table (Répliquer automatiquement de nouvelles colonnes ajoutées à la table source) lorsque vous créez ou modifiez les propriétés d'un abonnement Q dans le Centre de réplication. Pour plus d'informations, consultez ces rubriques :</p> <ul style="list-style-type: none"> • z/OS Enabling replication of ADD COLUMN and SET DATA TYPE operations (Activation de la réplication des opérations ADD COLUMN et SET DATA TYPE) • «Activation d'une réplication automatique des colonnes nouvellement ajoutées à partir de la table source», à la page 140 <p>Versions antérieures Utilisez le Tableau de bord de réplication Q ou la commande ASNCLP ALTER ADD COLUMN ou insérez manuellement un signal ADDCOL pour indiquer que vous voulez répliquer la nouvelle colonne.</p>
<p>TRUNCATE TABLE</p>	<p>z/OS Une opération TRUNCATE est consignée similairement à une suppression globale, de sorte que l'opération est répliquée en tant que série de suppressions à lignes uniques.</p> <p>Linux UNIX Windows La réplication des opérations TRUNCATE n'est pas prise en charge.</p>	<p>z/OS Aucune action n'est requise. Si la table cible possède des lignes qui ne sont pas dans la table source, ces lignes ne sont pas supprimées.</p> <p>Linux UNIX Windows Si vous avez besoin d'effectuer une opération TRUNCATE sur une table cible en plus de sa source, vous devez émettre l'instruction TRUNCATE directement par rapport à la table cible.</p>

Tableau 25. Gestion des changements DDL à la base de données source par Q Capture et mode d'action (suite)

Opération de langage de définition de données	Comment elle est traitée	Action à accomplir
ALTER TABLE ALTER COLUMN SET DATA TYPE	Répliqué automatiquement pour la version 10.1 et ultérieure. Le type de données de la colonne de la table cible correspondante est modifié et la réplication se poursuit normalement.	<p>z/OS</p> <p>Voir Enabling replication of ADD COLUMN and SET DATA TYPE operations (Activation de la réplication des opérations ADD COLUMN et SET DATA TYPE).</p> <p>Linux UNIX Windows</p> <p>Voir «Réplication automatique des opérations ALTER TABLE ALTER COLUMN SET DATA TYPE», à la page 146.</p>
Autre DDL qui modifie la structure d'une table	Q Capture laisse l'abonnement Q inchangé.	<ol style="list-style-type: none"> 1. Arrêtez l'abonnement Q. 2. Modifiez les tables source et cible. 3. Démarrez l'abonnement Q.
Langage de définition de données qui n'altère pas la structure d'une table Par exemple : • CREATE INDEX • ALTER FOREIGN KEY • ADD CONSTRAINT	Q Capture laisse l'abonnement Q actif.	Vérifiez que les contraintes d'unicité, les clés principales et les contraintes d'intégrité référentielle correspondent entre les tables source et cible. Si vous modifiez l'une de ces propriétés sur la table source, effectuez la modification correspondante sur la table cible pour éviter un comportement imprévu. Redémarrez également le programme Q Apply pour qu'il prenne en compte la modification.

Modification des propriétés des mappes de files de réplication

Une mappe de files de réplication comprend des options concernant le traitement des abonnements Q utilisant une file d'attente d'envoi et une file de réception couplées. En mettant à jour une mappe de file d'attente et en réinitialisant ensuite la file d'attente d'envoi, la file d'attente de réception ou les deux, vous pouvez modifier certains de ces paramètres sans arrêter la réplication.

A propos de cette tâche

Les propriétés des mappes de file d'attente de réplication sont enregistrées dans les tables de contrôle. Lorsque vous réinitialisez la file d'attente d'envoi, le programme Q Capture obtient les derniers paramètres à partir de la table IBMQREP_SENDQUEUES. Lorsque vous réinitialisez une file d'attente de réception, le programme Q Apply obtient les derniers paramètres à partir de la table IBMQREP_RECVQUEUES. Les nouveaux paramètres affectent tous les abonnements Q qui utilisent la mappe de files de réplication.

Si le programme Q Capture est arrêté lorsque vous faites des modifications sur la mappe de files d'attente, vous n'avez pas besoin de réinitialiser la file d'attente d'envoi car Q Capture lit les modifications lorsqu'il démarre. Si le programme Q Apply est arrêté lorsque vous faites des modifications, vous n'avez pas besoin de réinitialiser la file d'attente de réception.

Procédure

Pour mettre à jour une mappe de file d'attente de réplication et demander au programme Q Capture ou au programme Q Apply de reconnaître les modifications :

1. Modifiez les propriétés de la mappe de file d'attente en utilisant l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande ALTER REPLQMAP . Par exemple, les commandes suivantes définissent l'environnement et modifient la mappe de file d'attente de réplication SAMPLE_ASN1_TO_TARGETDB_ASN1 en définissant le nombre d'unités d'exécution de l'agent Q Apply sur quatre et en spécifiant que le programme Q Capture devrait s'arrêter si une erreur se produit dans une file d'attente : ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE ASN1; SET SERVER TARGET TO DB TARGETDB; SET APPLY SCHEMA ASN1; SET RUN SCRIPT LATER; ALTER REPLQMAP SAMPLE_ASN1_TO_TARGET_ASN1 USING NUM APPLY AGENTS 4 ERROR ACTION S;
Centre de réplication	Utilisez la fenêtre Propriétés de la mappe de files de réplication. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur une mappe de files de réplication et sélectionnez Propriétés .
Le tableau de bord de la réplication Q	Utilisez la page Files d'attente. Sélectionnez la paire programme Q Capture - programme Q Apply puis, dans l'onglet Propriétés, effectuez vos modifications et cliquez sur Enregistrer .

2. Si vous modifiez les files d'attente WebSphere MQ dans la mappe de files d'attente, utilisez les outils d'administration de réplication pour valider toute nouvelle file d'attente et tester le flux de messages entre les files d'attente.
3. Réinitialisez la file d'attente d'envoi, de réception ou les deux en fonction des propriétés que vous avez modifiées. Si vous devez réinitialiser les deux files d'attente, vous pouvez le faire dans n'importe quel ordre. Choisissez l'une des méthodes suivantes :

Méthode	Description
Centre de réplication (file d'attente d'envoi)	Utilisez la fenêtre Gérer les files d'attente d'envoi. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture où la file d'attente d'envoi se trouve et sélectionnez Gérer > Files d'attente d'envoi .
Centre de réplication (file d'attente de réception)	Utilisez la fenêtre Gérer les files d'attente de réception. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Apply où la file d'attente de réception se trouve et sélectionnez Gérer > Files d'attente de réception .

Méthode	Description
Paramètre reinitq (file d'attente d'envoi)	<p>Associez ce paramètre à la commande MODIFY sous z/OS ou à la commande asnqccmd sous Linux, UNIX, Windows, et les services système UNIX (USS) sous z/OS pour réinitialiser une file d'attente d'envoi dont vous avez modifié les propriétés dans la mappe de files d'attente :</p> <p>MODIFY f myqcap, reinitq=<i>nom.file</i></p> <p>Où myqcap est le nom du travail Q Capture.</p> <p>asnqccmd asnqccmd capture_server=<i>nom_serveur</i> schéma_capture=schema reinitq=<i>nom_file_attente</i></p> <p>Où <i>nom_serveur</i> est le nom de la base de données ou du sous-système où le programme Q Capture est exécuté, <i>schéma_capture</i> identifie le programme Q Capture qui utilise la file d'attente d'envoi et <i>nom_file_attente</i> est le nom de la file d'attente d'envoi que vous souhaitez réinitialiser.</p>
Paramètre reinitq (file d'attente de réception)	<p>Associez ce paramètre à la commande MODIFY sous z/OS ou à la commande asnqacmd sous Linux, UNIX, Windows et les services système UNIX (USS) sous z/OS pour réinitialiser une file d'attente de réception dont vous avez modifié les propriétés dans la mappe de files d'attente :</p> <p>MODIFY f myqapp, reinitq=<i>nom_file_attente</i></p> <p>Où myqapp correspond au nom du travail Q Apply.</p> <p>asnqacmd asnqacmd apply_server=<i>nom_serveur</i> apply_schema=<i>nom_schéma</i> reinitq=<i>nom_file</i></p> <p>Où <i>nom_serveur</i> est le nom de la base de données ou du sous-système où s'exécute le programme Q Apply, <i>nom_schéma</i> identifie le programme Q Apply qui utilise la file d'attente de réception et <i>nom_file</i> est le nom de la file d'attente de réception que vous souhaitez réinitialiser.</p>

Modification des noms de file d'attente lorsque la mappe de files d'attente est utilisée par les abonnements Q

Si une mappe de files de réplication fait déjà partie d'un abonnement Q, vous devez mettre à jour manuellement les tables de contrôle si vous voulez modifier le nom de la file d'attente d'envoi ou la file d'attente de réception dans la mappe de files d'attente.

A propos de cette tâche

Si une mappe de files d'attente n'est pas utilisée pour des abonnements Q, vous pouvez toujours utiliser le Centre de réplication ou le programme ASNCLP pour modifier les noms de file d'attente dans la mappe de files d'attente.

Procédure

1. Arrêtez le programme Q Capture et laissez le programme Q Apply traiter tous les messages dans la file d'attente de réception.
2. Facultatif : Si des abonnements Q utilisant la mappe de files d'attente sont actifs, arrêtez le programme Q Apply. Cette étape n'est pas nécessaire si vous voulez modifier uniquement le nom de la file d'attente d'envoi.
3. Facultatif : Si vous avez créé des tables de contrôle Q Capture à l'aide des outils d'administration de réplication avant la Version 9.7 groupe de correctifs 2, utilisez la commande suivante pour supprimer une contrainte de clé étrangère de la table IBMQREP_SUBS qui fait référence à la table IBMQREP_SENDQUEUEES :


```
ALTER TABLE schéma.IBMQREP_SUBS DROP CONSTRAINT FKSENDQ
```


 Cette contrainte ne fait plus partie des tables de contrôle qui sont créées en démarrant avec la Version 9.7 groupe de correctifs 2.
4. Utilisez SQL pour modifier le nom de la file d'attente d'envoi dans la table IBMQREP_SENDQUEUEES. Généralement, la file d'attente d'envoi et la file d'attente de réception correspondante ont le même nom. Si tel est le cas, veillez à changer également le nom de la file d'attente de réception.

Par exemple, l'instruction suivante modifier le nom des files d'attente d'envoi et de réception de Q2 en Q3 dans la table IBMQREP_SENDQUEUEES (dans une étape ultérieure, le nom de la file d'attente de réception est modifié dans les tables de contrôle Q Apply):

```
UPDATE schéma.IBMQREP_SENDQUEUEES SET SENDQ='Q3',RECVQ='Q3' WHERE SENDQ='Q2'
```
5. Mettez à jour le nom de la file d'attente d'envoi dans la table IBMQREP_SUBS. Par exemple :


```
UPDATE schéma.IBMQREP_SUBS SET SENDQ='Q3' WHERE SUBNAME='EMPLOYEE0001';
```
6. Sur le serveur Q Apply, mettez à jour les noms de file d'attente dans trois tables de contrôle : IBMQREP_RECVQUEUEES, IBMQREP_TARGETS, et IBMQREP_TRG_COLS. Par exemple :


```
UPDATE schéma.IBMQREP_RECVQUEUEES SET RECVQ='Q3',SENDQ='Q3' WHERE RECVQ='Q2'
UPDATE schéma.IBMQREP_TARGETS SET RECVQ='Q3' WHERE RECVQ='Q2'
UPDATE schéma.IBMQREP_TRG_COLS SET RECVQ='Q3' WHERE RECVQ='Q2'
```
7. Facultatif : Si vous avez annulé la clé externe dans l'étape 2, restaurez-la à l'aide de la commande suivante :


```
ALTER TABLE schéma.IBMQREP_SUBS ADD CONSTRAINT FKSENDQ FOREIGN KEY(SENDQ)
REFERENCES schéma.IBMQREP_SENDQUEUEES(SENDQ)
```
8.  Facultatif : Si vous avez supprimé et restauré la clé externe, exécutez l'utilitaire CHECK DATA pour amener l'espace de table qui contient la table IBMQREP_SUBS en dehors de l'état CHECK-pending.
9. Facultatif : Si vous avez annulé le programme Q Apply au cours de l'étape 2, démarrez le programme Q Apply.
10. Démarrez le programme Q Capture en mode à chaud.

Redémarrage des abonnements Q échoués sans les supprimer ni les recréer

Lorsque des abonnements Q ne parviennent pas à démarrer et laissent les tables de contrôle dans un état incohérent, vous pouvez réinitialiser manuellement l'état de l'abonnement Q sans avoir à supprimer ou à recréer les abonnements Q.

A propos de cette tâche

Cette procédure est particulièrement utile pour la réplication bidirectionnelle ou entre homologues.

Par exemple, il se peut que les abonnements Q ne puissent pas démarrer parce qu'un canal WebSphere MQ n'a pas été démarré ou que la configuration de WebSphere MQ n'a pas été définie correctement. Dans ces cas, l'abonnement Q est correctement défini, mais le message du schéma d'abonnement de Q Capture à Q Apply n'est pas transmis.

Cet incident peut provoquer des états d'abonnements Q incohérents entre les tables de contrôle source Q Capture et cible Q Apply. Vous pouvez recevoir un message d'erreur pour lequel la section "Réponse de l'utilisateur" recommande que vous supprimiez ou recréiez les abonnements Q échoués après avoir corrigé le problème de WebSphere MQ.

Vous pouvez éviter de supprimer ou de recréer les abonnements Q en réinitialisant leur état dans les tables de contrôle. La procédure ci-dessous garantit que les abonnements Q sont effacés (retour à l'état inactif ou I) avant d'être redémarrés, comme au moment de leur création pour la première fois.

Procédure

Pour redémarrer les abonnements Q échoués sans les supprimer ou les recréer :

1. Traitez le problème qui a empêché le démarrage des abonnements Q. Il se peut que vous deviez mettre à niveau le logiciel ou effectuer des modifications mineures sur les tables source, les tables cible ou les deux, ou modifier les éléments de base de données associés. Utilisez tout code anomalie ou informations de réponse contenues dans les messages d'erreur pour vous aider à diagnostiquer le problème.
2. Réinitialisez les marqueurs de statut dans les tables de contrôle en exécutant les instructions SQL suivantes :

Serveur Q Capture

```
UPDATE schéma.IBMQREP_SUBS
SET STATE='I', STATE_TRANSITION=NULL,
GROUP_MEMBERS=NULL WHERE SUBNAME='nom_abonnement_Q'
```

Serveur Q Apply

```
UPDATE schéma.IBMQREP_TARGETS
SET STATE='I' WHERE SUBNAME='nom_abonnement_Q'
```

3. Démarrez les abonnements Q.

Placer des abonnements Q dans un spill mode à des fins de maintenance

Vous pouvez placer des abonnements Q dans un mode déchargement temporaire afin d'effectuer la maintenance des tables cible sans arrêter la capture des modifications à la source ou sans affecter la réplication sur les autres tables.

A propos de cette tâche

Lorsque vous placez un abonnement Q en mode déchargement, les modifications ne sont pas appliquées à la table cible mais sont déchargées sur une file d'attente auxiliaire créée dynamiquement. Les modifications continuent d'être saisies et appliquées pour les autres abonnements Q. Une fois la maintenance terminée, vous pouvez envoyer une commande de reprise au programme Q Apply.

La file d'attente auxiliaire temporaire est créée en fonction de la définition de la file d'attente du modèle utilisée pour les opérations de chargement. Assurez-vous que la longueur maximale de la file d'attente est suffisamment grande pour conserver les lignes déversées jusqu'à ce qu'elles puissent être appliquées après la reprise des opérations.

Remarque sur la cohérence transactionnelle : Si vous choisissez de télécharger les données répliquées depuis une table, sachez que toute transaction impliquant cette table et les autres tables peuvent devenir incohérentes en mode déchargement. Par exemple, supposez que vous vouliez effectuer la maintenance sur la table cible T1, et que vous lanciez la commande **spillsub**. Considérez une transaction source qui met à jour les tables T1 et T2 :

```
UPDATE T1;  
UPDATE T2;  
COMMIT;
```

L'opération UPDATE T1 est déchargée, mais l'opération UPDATE T2 est appliquée à la table cible, et une validation de base de données et une validation WebSphere MQ sont émises. Si vous regardez les tables cible, seule la mise à jour de T2 existe. Ainsi, une transaction partielle a été appliquée et validée dans la base de données, et la cohérence transactionnelle n'a pas été respectée. Si vous souhaitez une cohérence transactionnelle à tout moment, vous ne devez pas utiliser la fonction **spillsub**.

Restrictions

Le mode Déchargement n'est pas autorisé si l'abonnement Q implique une table qui possède des dépendances d'intégrité référentielle. Vous devez supprimer et restaurer les contraintes de RI sur la table cible.

Recommandation : Utilisez la même file d'attente de réception pour tous les abonnements Q qui impliquent des tables liées à RI. Si vous devez effectuer une maintenance, arrêtez le traitement de messages dans la file d'attente de réception au lieu de télécharger un abonnement Q spécifique.

Procédure

1. Placez l'abonnement Q en mode déchargement à l'aide du paramètre **spillsub** avec la commande **MODIFY** ou la commande **asnqacmd** :

```
F nom_travail_Q_Apply,spillsub="nom_file-attente-réception:nom_abonnement_Q"
```

2. Pour reprendre les opérations normales, utilisez le paramètre **resumesub** :

```
F nom_travail_Q_Apply,resumesub="nom_file_attente_réception:nom_abonnement_Q"
```

Lorsque toutes les lignes de la file d'attente auxiliaire sont appliquées, le programme Q Apply replace l'abonnement Q à l'état actif (A).

Suppression d'abonnements Q

Vous pouvez supprimer un abonnement Q qui n'est pas traité de manière active.

Avant de commencer

L'abonnement Q que vous voulez supprimer doit avoir pour état I (inactif) ou N (nouveau).

A propos de cette tâche

Lorsque vous supprimez un abonnement Q, vous avez l'option de déplacer la table à laquelle il renvoie. Vous pouvez également déplacer l'espace table pour la table cible.

Si la table cible est dans une base de données non-DB2, vous pouvez indiquer s'il faut supprimer l'alias pour la table cible lorsque l'abonnement Q est supprimé.

Remarque : Supprimer un abonnement Q ne supprime pas la mappe de files de réplication qu'il utilise.

Procédure

Pour supprimer un ou plusieurs abonnements Q, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande DROP QSUB. Par exemple, les commandes suivantes définissent l'environnement et déplacent l'abonnement Q pour une réplication unidirectionnelle EMPLOYEE0001 :</p> <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET SERVER TARGET TO DB TARGET;</pre> <p>DROP QSUB (SUBNAME EMPLOYEE0001 USING REPLQMAP SAMPLE_ASN1_TO_TARGETDB_ASN1);</p> <p>Utilisez la commande SET DROP pour préciser s'il faut déplacer la table cible et son espace table.</p>
Centre de réplication	<p>Utilisez la fenêtre Supprimer les abonnements Q. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur un abonnement Q puis sélectionnez Supprimer.</p> <p>Vous pouvez utiliser cette fenêtre à titre facultatif pour déplacer les tables cible des abonnements Q que vous supprimez. Si vous déplacez une table cible, vous pouvez utiliser cette fenêtre à titre facultatif pour déplacer l'espace table associé si aucune autre table n'utilise ce dernier.</p>

Suppression des mappes de files de réplication

Vous pouvez supprimer une mappe de files de réplication qui n'est plus nécessaire pour aucun des abonnements Q.

Procédure

Pour supprimer une mappe de files de réplication :

1. Assurez-vous qu'aucun abonnement Q n'utilise la mappe de files de réplication.
 - a. **Facultatif** : Utilisez la fenêtre Afficher les relations du Centre de réplication pour voir si des abonnements Q utilisent cette mappe de files de réplication. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur la mappe de file d'attente et sélectionnez **Afficher les relations**.
 - b. Si des abonnements Q utilisent la mappe de file d'attente, supprimez-les.
2. Supprimer la mappe de file d'attente. Utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande DROP REPLQMAP . Par exemple, les commandes suivantes définissent l'environnement et suppriment la mappe de file d'attente SAMPLE_ASN1_TO_TARGET_ASN1 : <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE ASN1; SET SERVER TARGET TO DB TARGET; SET APPLY SCHEMA ASN1; SET RUN SCRIPT LATER;</pre> <pre>DROP REPLQMAP SAMPLE_ASN1_TO_TARGETDB_ASN1;</pre>
Centre de réplication	Utilisez la fenêtre Supprimer les mappes de files de réplication. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur la mappe de files de réplication et sélectionnez Supprimer .

Suppression des tables de contrôle Q Capture ou Q Apply

Lorsque vous supprimez les tables de contrôle Q Capture ou Q Apply, vous supprimez également les schémas Q Capture ou Q Apply associés depuis l'arborescence des objets du Centre de réplication. Le fait de supprimer les tables de contrôle supprime également les instances du programme Q Capture ou Q Apply.

A propos de cette tâche

La suppression des tables de contrôle agit également sur les objets suivants :

Tables de contrôle Q Capture

Les abonnements Q et les publications sont supprimés car les définitions de ces deux objets sont archivées dans les mêmes tables de contrôle. La suppression des tables de contrôle agit également sur les mappes de files de réplication et de publication.

Tables de contrôle Q Apply

Les abonnements Q et les mappes de files de réplication sont supprimés. Tous les pseudonymes créés pour les tables cible non DB2 sont également supprimés.

Les abonnements Q et les mappes de files de réplication sont définis sur les serveurs Q Capture et Q Apply. Si vous ne supprimez que les tables de contrôle Q Capture ou Q Apply, tous les abonnements Q ou mappes de files de réplication définis également dans les tables de contrôle sur le serveur opposé doivent être supprimés avant que vous ne supprimiez les tables de contrôle.

Procédure

Pour supprimer les tables de contrôle Q Capture ou Q Apply :

1. Arrêtez le programme Q Capture ou Q Apply qui utilise les tables de contrôle que vous voulez supprimer.
2. Si vous supprimez que les tables de contrôle Q Capture ou que celles de Q Apply, procédez comme suit :
 - Désactivez tous les abonnements Q qui sont également définis sur le serveur opposé.
 - Supprimez les abonnements Q.

- Supprimez toutes les mappes de files de réplication qui sont également définies sur le serveur opposé.
3. Appliquez l'une des méthodes suivantes pour supprimer les tables de contrôle :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande DROP CONTROL TABLES ON. Par exemple, les commandes suivantes définissent l'environnement et suppriment les tables de contrôle sur le serveur Q Apply TARGET identifiées par le schéma ASN1 : ASNCLP SESSION SET TO Q REPLICATION; SET SERVER APPLY TO DB TARGET; SET APPLY SCHEMA ASN1; DROP CONTROL TABLES ON APPLY SERVER;
Centre de réplication	Utilisez la fenêtre Supprimer le schéma Q Capture ou Supprimer le schéma Q Apply. Pour ouvrir les fenêtres, cliquez avec le bouton droit de la souris sur le schéma et sélectionnez Drop . Si vous supprimez le dernier schéma sur un serveur Q Capture ou Q Apply, ce serveur ne contiendra plus d'ensemble de tables de contrôle et sera supprimé de l'arborescence des objets du Centre de réplication.

Chapitre 18. Modification d'un environnement de publication d'événements

Vous pouvez modifier les propriétés d'objets de publication d'événements (tels que des publications et des mappes de files pour la publication) après les avoir définis, sans interrompre la publication dans la plupart des cas.

Modification des propriétés de publications

Vous pouvez modifier les propriétés de publications sans interrompre la publication. Dans un premier temps, vous faites vos changements. Réinitialisez ensuite soit une publication XML unique, soit le programme Q Capture si vous modifiez plusieurs publications.

Restrictions

Il n'est pas possible de modifier toutes les propriétés des publications XML.

- Si vous utilisez le programme de ligne de commande ASNCLP pour apporter des modifications, vérifiez la syntaxe de la commande **ALTER PUB** afin de déterminer les propriétés modifiables.
- Dans le Centre de réplication, les commandes correspondant aux propriétés que vous ne pouvez pas modifier sont désactivées dans le bloc-notes Propriétés de la publication XML.

A propos de cette tâche

Si vous réinitialisez un programme Q Capture, cette opération entraîne également la réinitialisation des abonnements Q définis dans le schéma Q Capture. Une telle réinitialisation requiert également davantage de ressources système. Si vous ne souhaitez modifier que les attributs d'une ou de deux publications XML, il est moins coûteux en termes de ressources de réinitialiser une publication XML à la fois.

Procédure

Pour modifier les propriétés de publications XML sans interrompre la publication :

1. Utilisez l'une des méthodes suivantes pour modifier la publication XML :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande ALTER PUB . Par exemple, les commandes suivantes définissent l'environnement et modifient la publication EMPLOYEE0001 en envoyant toutes les colonnes d'une ligne qui font partie de la publication dès que l'une d'entre elles a été modifiée et n'envoient pas les lignes supprimées : ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE EP1; SET RUN SCRIPT LATER; ALTER PUB EMPLOYEE0001 FOR JK.EMPLOYEE OPTIONS CHANGED COLS ONLY N SUPPRESS DELETES Y;

Méthode	Description
Centre de réplication	Utilisez le bloc-notes Propriétés de la publication XML. Pour ouvrir ce bloc-notes, cliquez avec le bouton droit de la souris sur une publication XML, puis sélectionnez Propriétés .

2. Si vous modifiez une seule publication XML, réinitialisez-la en utilisant l'une des méthodes suivantes :

Option	Description
Centre de réplication	Utilisez la fenêtre Gestion des publications. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture où se trouve la table source de la publication XML, puis sélectionnez Gestion > des publications XML .
SQL	<p>Insérez un signal REINIT_SUB dans la table IBMQREP_SIGNAL au niveau du serveur Q Capture :</p> <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'REINIT_SUB', 'nompub', 'P');</pre> <p>Où <i>schéma</i> identifie le programme Q Capture qui traite la publication XML et <i>nompub</i> correspond au nom de la publication XML que vous souhaitez réinitialiser.</p>

3. Si vous modifiez plusieurs publications XML, réinitialisez le programme Q Capture en utilisant l'une des méthodes suivantes :

Méthode	Description
Paramètre reinit	<p>Associez ce paramètre à la commande MODIFY sous z/OS ou à la commande asqccmd sous Linux, UNIX, Windows et les services systèmes UNIX (USS) sous z/OS pour réinitialiser toutes les publications d'un schéma Q Capture :</p> <p>MODIFY</p> <pre>f myqcap,reinit</pre> <p>Où myqcap est le nom du travail Q Capture.</p> <p>asqccmd</p> <pre>asqccmd capture_server=nom_serveur capture_schema=schéma reinit</pre> <p>Où <i>nom_serveur</i> correspond au nom de la base de données ou du sous-système où le programme Q Capture est exécuté, et <i>schéma</i> identifie le programme Q Capture dont vous souhaitez réinitialiser toutes les publications XML.</p>
Centre de réplication	Utilisez la fenêtre Réinitialiser Q Capture. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture contenant le programme Q Capture que vous souhaitez réinitialiser puis sélectionnez Réinitialiser le programme Q Capture .

Ajout de colonnes aux publications existantes

Vous pouvez ajouter des colonnes dynamiquement à une publication existante tout en continuant à publier des modifications à partir de la table source.

Avant de commencer

- La publication XML à laquelle les colonnes sont ajoutées doit être à l'état A (actif) lorsque le signal est inséré.
- Si le type de données de la colonne est LONG VARCHAR ou GRAPHIC, la base de données ou le sous-système source doit être configuré avec DATA CAPTURE CHANGES INCLUDE VARCHAR COLUMNS.

Restrictions

- Les colonnes que vous ajoutez doivent admettre la valeur NULL, ou être définies comme NOT NULL WITH DEFAULT.
- Vous ne pouvez pas ajouter plus de 20 colonnes au cours d'un intervalle de validation WebSphere MQ unique, qui est défini par le paramètre Q Capture COMMIT_INTERVAL.

A propos de cette tâche

Les colonnes peuvent déjà exister dans la table source, ou vous pouvez les ajouter à la table puis à la publication XML au cours de la même transaction.

Pour ajouter une nouvelle colonne, insérez un signal SQL sur le serveur Q Capture. Le signal SQL contient des détails sur la colonne. Si vous souhaitez ajouter plusieurs colonnes à une publication, vous insérez un signal pour chaque nouvelle colonne. Vous pouvez ajouter plusieurs colonnes au cours d'une seule transaction.

Vous pouvez ajouter des colonnes à toutes les publications qui souscrivent à une table source en spécifiant l'option **SOURCE** au lieu de l'option **PUB**.

Conseil : Si vous prévoyez de mettre au repos la base de données ou l'instance source avant d'ajouter des colonnes, vous pouvez utiliser le Centre de réplication ou la commande **asnccmd chgparms** pour attribuer la valeur N (no) au paramètre TERM du programme Q Capture. La définition de TERM=N indique au programme de poursuivre son exécution pendant que la base de données ou l'instance est en mode repos. Lorsque DB2 quitte le mode repos, le programme Q reprend la capture des modifications à partir du dernier point de redémarrage dans le journal sans vous demander de redémarrer le programme.

Procédure

Pour ajouter des colonnes à une publication existante, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande ALTER ADD COLUMN . Par exemple, la commande suivante ajoute la colonne BONUS à la publication DEPARTMENT0001 : ALTER ADD COLUMN USING SIGNAL (BONUS) PUB DEPARTMENT0001;

Méthode	Description
SQL	<p>Utilisez une invite de commande ou l'un des outils de ligne de commande DB2 pour insérer un signal ADDCOL dans la table IBMQREP_SIGNAL sur le serveur Q Capture. Par exemple :</p> <pre>insert into schéma.IBMQREP_SIGNAL(SIGNAL_TIME, SIGNAL_TYPE, SIGNAL_SUBTYPE, SIGNAL_INPUT_IN, SIGNAL_STATE) values (CURRENT_TIMESTAMP, 'CMD', 'ADDCOL', 'nompub;nom_colonne', 'P');</pre> <p><i>schéma</i> Identifie le programme Q Capture qui traite la publication à laquelle vous ajoutez une colonne.</p> <p><i>nompub;nom_colonne</i> Nom de la publication à laquelle vous ajoutez la colonne et nom de la colonne que vous ajoutez, séparés par un point-virgule. Ces noms font la distinction entre les majuscules et les minuscules et ne requièrent pas de guillemets pour préserver la casse.</p>

Après avoir traité le signal, le programme Q Capture envoie un message d'ajout de colonne à l'application utilisateur et commence à capturer les modifications apportées à la nouvelle colonne lors de la lecture des données de journal qui incluent la colonne. Les modifications apportées à la colonne sont validées après publication de la validation de l'insertion du signal ADDCOL.

Suppression de publications

Vous pouvez supprimer une publication XML qui n'est pas activement traitée par un programme Q Capture.

Avant de commencer

La publication XML à supprimer doit être à l'état I (inactif) ou N (nouveau).

A propos de cette tâche

La suppression d'une publication XML n'entraîne pas celle de la mappe de files pour la publication qu'elle utilise.

Procédure

Pour supprimer une ou plusieurs publications XML, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande DROP PUB. Par exemples, les commandes suivantes définissent l'environnement et suppriment la publication EMPLOYEE0001 : <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE EP1; SET RUN SCRIPT LATER; DROP PUB (PUBNAME EMPLOYEE0001);</pre>
Centre de réplication	Utilisez la fenêtre Suppression de publications XML. Pour ouvrir cette fenêtre, cliquez avec le bouton droit de la souris sur une publication XML, puis sélectionnez Supprimer .

Modification des propriétés des mappes de files pour la publication

Une mappe de files pour la publication pointe vers une file d'attente d'envoi et inclut les paramètres de gestion par un programme Q Capture des publications qui empruntent la file d'attente d'envoi. En mettant à jour une mappe de files d'attente puis en réinitialisant la file d'attente d'envoi, vous pouvez modifier certains de ces paramètres sans qu'il soit nécessaire d'arrêter la publication des modifications à partir de la table source.

A propos de cette tâche

Les propriétés des mappes de files pour la publication sont enregistrées dans la table de contrôle IBMQREP_SENDQUEUES. Lorsque vous réinitialisez une file d'attente d'envoi, le programme Q Capture obtient les derniers paramètres de cette table. Les nouveaux paramètres affectent toutes les publications qui empruntent la file d'attente d'envoi.

Procédure

Pour mettre à jour une mappe de files d'attente pour la publication sans arrêter la publication :

1. Modifiez les propriétés de la mappe de file d'attente en utilisant l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande ALTER PUBQMAP. Par exemple, les commandes suivantes définissent l'environnement et modifient la mappe de files pour la publication SAMPLE_EP1_TO_SUBSCRIBER en transformant le type de message de ligne en transaction et en spécifiant six secondes entre les messages de pulsation : <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE EP1; SET RUN SCRIPT LATER; ALTER PUBQMAP SAMPLE_EP1_TO_SUBSCRIBER USING MESSAGE CONTENT TYPE T HEARTBEAT INTERVAL 6;</pre>

Méthode	Description
Centre de réplication	Utilisez la fenêtre Propriétés de la mappe de files pour la publication. Pour ouvrir cette fenêtre, cliquez avec le bouton droit de la souris sur une mappe de files pour la publication et sélectionnez Propriétés .
Le tableau de bord de la réplication Q	Utilisez la page Files d'attente. Sélectionnez le programme Q Capture puis, dans l'onglet Propriétés, effectuez vos modifications et cliquez sur Enregistrer .

- Si vous modifiez la file d'attente WebSphere MQ utilisée comme file d'attente d'envoi, utilisez les outils d'administration de la réplication pour valider la nouvelle file d'attente.
- Utilisez l'une des méthodes suivantes pour inviter un programme Q Capture à reconnaître les modifications que vous avez effectuées sans arrêter le programme Q Capture :

Méthode	Description
Paramètre reinitq	<p>Associez ce paramètre à la commande MODIFY sous z/OS ou à la commande asnqccmd sous Linux, UNIX, Windows et les services systèmes UNIX (USS) sous z/OS pour réinitialiser une file d'attente d'envoi dont vous avez modifié les propriétés :</p> <p>MODIFY</p> <pre>f myqcap,reinitq=<i>nom.file</i></pre> <p>Où myqcap est le nom du travail Q Capture.</p> <p>asnqccmd</p> <pre>asnqccmd capture_server=<i>nom_serveur</i> schéma_capture=schema reinitq=<i>nom_file_attente</i></pre> <p>Où <i>nom_serveur</i> est le nom de la base de données ou du sous-système où le programme Q Capture est exécuté, <i>schéma_capture</i> identifie le programme Q Capture qui utilise la file d'attente d'envoi et <i>nom_file_attente</i> est le nom de la file d'attente d'envoi que vous souhaitez réinitialiser.</p>
Centre de réplication	Utilisez la fenêtre Gérer les files d'attente d'envoi. Pour ouvrir cette fenêtre, cliquez avec le bouton droit de la souris sur le serveur Q Capture sur lequel se trouve la file d'attente d'envoi que vous souhaitez réinitialiser et sélectionnez Gestion > des files d'attente d'envoi .

Suppression des mappes de files de réplication

Vous pouvez supprimer une mappe de files pour la publication dont aucune publication n'a plus besoin.

Procédure

Pour supprimer une mappe de files pour la publication :

- Assurez-vous qu'aucune publication n'utilise la mappe.
 - Facultatif : Ouvrez la fenêtre Affichage des objets associés dans le Centre de réplication pour voir si des publications utilisent la mappe de files pour

la publication. Pour ouvrir cette fenêtre, cliquez avec le bouton droit de la souris sur la mappe de files pour la publication et sélectionnez **Affichage des objets associés**.

- b. Si des publications utilisent la mappe, supprimez-les.
2. Supprimez ensuite la mappe de files pour la publication. Utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande DROP PUBQMAP. Par exemple, les commandes suivantes définissent l'environnement et suppriment la mappe de files pour la publication SAMPLE_EP1_TO_SUBSCRIBER : <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE EP1; SET RUN SCRIPT LATER; DROP PUBQMAP SAMPLE_EP1_TO_SUBSCRIBER;</pre>
Centre de réplication	Utilisez la fenêtre Suppression de mappes de files d'attente pour la publication. Pour ouvrir cette fenêtre, cliquez avec le bouton droit de la souris sur la mappe de files pour la publication et sélectionnez Suppression .

Suppression des tables de contrôle Q Capture

Lorsque vous supprimez des tables de contrôle Q Capture, vous retirez également les publications, abonnements Q, mappes de files d'attente pour la publication ou mappes de files d'attente pour la réplication qui sont définis dans ces tables de contrôle.

Avant de commencer

- Le programme Q Capture qui utilise les tables de contrôle doit être arrêté.
- Les abonnements Q ou mappes de files pour la réplication contenus dans les tables de contrôle Q sont également définis dans les tables de contrôle Q Apply. Vous devez supprimer les abonnements Q ou mappes de files avant de supprimer les tables de contrôle Q Capture.

A propos de cette tâche

La suppression des tables de contrôle Q Capture entraîne également le retrait du schéma Q Capture associé de l'arborescence des objets du Centre de réplication, ainsi que le retrait de l'instance du programme Q Capture associée.

Procédure

Pour supprimer les tables de contrôle Q Capture :

1. Arrêtez le programme Q Capture qui utilise les tables de contrôle que vous souhaitez supprimer.
2. Appliquez l'une des méthodes suivantes pour supprimer les tables de contrôle :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande DROP CONTROL TABLES ON. Par exemple, les commandes suivantes définissent l'environnement et suppriment sur le serveur Q Capture SAMPLE les tables de contrôle qui sont identifiées par le schéma ASN1 :</p> <pre>ASNCLP SESSION SET TO Q REPLICATION; SET SERVER CAPTURE TO DB SAMPLE; SET CAPTURE SCHEMA SOURCE ASN1; DROP CONTROL TABLES ON CAPTURE SERVER;</pre>
Centre de réplication	<p>Utilisez la fenêtre Suppression du schéma Q Capture. Pour ouvrir cette fenêtre, cliquez avec le bouton droit de la souris sur le schéma et sélectionnez Suppression.</p> <p>Si vous supprimez le dernier schéma sur un serveur Q Capture, ce serveur ne contient plus aucun ensemble de tables de contrôle et est supprimé de l'arborescence des objets du Centre de réplication.</p>

Chapitre 19. Vérification du statut des programmes de réplication Q et de publication d'événements

Les rubriques suivantes donnent des informations sur la manière dont vous pouvez vérifier le statut de votre environnement de réplication Q et de publication d'événements.

Vérification de l'état des programmes de réplication Q et de publication d'événements

Vous pouvez utiliser le Tableau de bord de la réplication Q ou les commandes système pour vérifier l'état actuel des programmes Q Capture et Q Apply. Vous pouvez utiliser une commande système pour vérifier l'état actuel du moniteur d'alertes de réplication.

A propos de cette tâche

Le Tableau de bord de la réplication Q vous offre des vues dynamiques vous permettant de détecter les modifications d'état pour les programmes Q Capture et Q Apply ; vous pouvez en outre configurer des alertes par e-mail vous indiquant lorsque l'un des programmes s'arrête. Les commandes système vous permettent de visualiser les messages concernant l'état des unités d'exécution des programmes. Ces messages peuvent vous permettre de déterminer si les programmes fonctionnent correctement. Pour les programmes Q Capture et Q Apply, vous pouvez utiliser les commandes pour afficher des rapports plus détaillés concernant les opérations du programme en cours.

Procédure

Pour vérifier l'état des programmes de réplication Q et de publication d'événements :

Utilisez l'une des méthodes suivantes :

Tableau de bord de la réplication Q

Vous pouvez voir l'état des programmes Q Capture et Q Apply depuis l'onglet récapitulatif ou vous pouvez cliquer sur l'onglet Programmes afin d'obtenir des informations plus détaillées concernant l'état. Depuis l'onglet du gestionnaire d'alertes, vous pouvez activer les alertes par e-mail pour l'état des programmes Q Capture et Q Apply et le tableau de bord envoie un e-mail à un ou plusieurs contact(s) indiqué(s) en cas d'arrêt des programmes.

Ligne de commande

Q Capture

Associez le paramètre **status** à la commande **MODIFY** sous z/OS ou à la commande **asnqccmd** sous Linux, UNIX, Windows et les services système UNIX (USS) sous z/OS pour afficher le message indiquant l'état de chaque unité d'exécution Q Capture.

Pour afficher un statut plus détaillé sur le programme et ses opérations, utilisez le paramètre **show details**.

MODIFY

```
f myqcap,status show details
```

Où myqcap est le nom du travail Q Capture.

asnqccmd

```
asnqccmd capture_server=serveur  
capture_schema=schéma  
status show details
```

Où *nom_serveur* est le nom de la base de données ou sous-système où le programme Q Capture est en cours d'exécution et *schéma* identifie le programme Q Capture que vous souhaitez vérifier.

Q Apply

Associez le paramètre **status** à la commande **MODIFY** sous z/OS ou à la commande **asnqacmd** sous Linux, UNIX, Windows et les services système UNIX (USS) sous z/OS pour afficher les messages indiquant l'état de chaque unité d'exécution Q Apply.

Pour afficher un statut plus détaillé sur le programme et ses opérations, utilisez le paramètre **show details**.

MODIFY

```
f myqapp,status show details
```

Où myqapp correspond au nom du travail Q Apply.

asnqacmd

```
asnqacmd apply_server=serveur  
apply_schema=schéma  
status show details
```

Où *nom_serveur* est le nom de la base de données ou sous-système où le programme Q Apply est en cours d'exécution et *schéma* identifie le programme Q Apply que vous souhaitez vérifier.

moniteur d'alertes de réplication

Associez le paramètre **status** à la commande **MODIFY** sous z/OS ou à la commande **asnqmcmd** sous Linux, UNIX, Windows et les services système UNIX (USS) sous z/OS pour afficher les messages indiquant l'état de chaque unité d'exécution du programme du moniteur.

MODIFY

```
f mymon,status
```

Où mymon est le nom de travail du moniteur.

asnmcmd

```
asnmcmd monitor_server=serveur  
monitor_qual=qualificateur  
status show details
```

Où *nom_serveur* est le nom de la base de données ou du sous-système sur lequel le programme du moniteur est en cours d'exécution et où *qualificateur* identifie le programme du moniteur que vous voulez vérifier.

Unités d'exécution des programmes Q Capture, Q Apply et du moniteur d'alertes de réplication

Lorsque vous vérifiez l'état des programmes de réplication Q et de publication d'événements, vous pouvez savoir si les programmes fonctionnent correctement grâce aux messages que vous recevez à propos des unités d'exécution du programme.

La liste suivante affiche les unités d'exécution des programmes Q Capture, Q Apply et du moniteur d'alertes de réplication. Les abréviations situées après chaque nom d'unité d'exécution indiquent si l'unité d'exécution interagit avec la base de données ("DB") ou avec WebSphere MQ ("MQ") :

Programme Q Capture

Un programme Q Capture possède les unités d'exécution suivantes :

Lecture de journal (DB)

Pour DB2, lit le journal de récupération de base de données et capture les modifications pour les tables abonnées. Pour les sources Oracle, lance la fonctionnalité Oracle LogMiner, lit dans la vue V\$LOGMNR_CONTENTS afin de trouver des modifications pour les tables abonnées. Reconstitue les enregistrements du journal en tant que transactions en mémoire avant de les transmettre à l'unité d'exécution agent lorsque la validation d'une transaction est terminée.

Agent (DB ; MQ)

Reçoit les transactions terminées de l'unité d'exécution de lecture du journal, convertit les transactions en messages WebSphere MQ et place les messages dans les files d'attente d'envoi. Gère les informations de redémarrage du programme.

Administration (DB ; MQ)

Gère les messages de contrôle mis dans la file d'attente d'administration par le programme Q Apply ou une application utilisateur et inscrit les statistiques dans les tables de contrôle.

Elagage (DB)

Supprime les données obsolètes des tables de contrôle de Q Capture.

Détention de verrou (DB sur toutes les plateformes sauf z/OS)

Gère les signaux envoyés au programme Q Capture. Empêche les programmes Q Capture ayant le même schéma d'être exécutés sur un serveur.

Les unités d'exécution agent, de lecture de journal, d'administration et d'élagage sont généralement en cours de fonctionnement ou au repos. L'unité d'exécution de détention de verrou est toujours en attente. Si l'unité d'exécution agent est en cours de fonctionnement mais que les données n'ont pas encore été enregistrées, vous pouvez utiliser les fenêtres Tableau de bord de réplication Q et Messages Q Capture du Centre de réplication, ou la table IBMQREP_CAPTRACE pour rechercher des messages susceptibles d'expliquer pourquoi les données ne sont pas enregistrées. Sinon, vérifiez dans les tables IBMQREP_CAPMON ou IBMQREP_CAPQMON que les enregistrements de journal sont en cours de traitement.

programme Q Apply

Un programme Q Apply possède les unités d'exécution suivantes :

Navigateur (DB ; MQ)

Lit des messages de transaction provenant d'une file d'attente de réception, gère les dépendances entre les transactions et lance une ou plusieurs unités d'exécution agent. Le programme Q Apply lance une unité d'exécution de navigation pour chaque file d'attente de réception. Cette unité d'exécution est identifiée par les caractères BRxxxxx, où xxxxx indique le nombre d'unités d'exécution de navigation d'un programme Q Apply.

Agent (DB ; MQ)

Reconstitue les transactions en mémoire et les applique aux cibles. Vous définissez le nombre d'unités d'exécution d'agent qui seront utilisées pour un traitement parallèle des transactions lorsque vous créez une mappe de files de réplication.

Agent de déversement (DB ; MQ)

Créé sur demande lorsque les modifications de l'abonnement Q doivent être appliquées à partir d'une file d'attente auxiliaire. Cette unité d'exécution est créée y compris lorsqu'une table a terminé le chargement ou lorsqu'une commande **resumesub** est reçue. Reconstitue les transactions contenues dans la file d'attente auxiliaire et les applique aux cibles. Les agents de déversement se terminent après le vidage de la file d'attente auxiliaire et l'activation de l'abonnement Q.

Administration (DB ; MQ)

Gère les tables de contrôle Q Apply. Démarre de nouveaux navigateurs lorsque la commande **startq** est reçue.

Moniteur (DB ; MQ)

Enregistre les informations relatives aux performances du programme Q Apply dans la table de contrôle IBMQREP_APPLYMON.

Elagage (DB ; MQ)

Supprime les lignes de la table de contrôle IBMQREP_DONEMSG une fois que les messages auxquels les lignes appartiennent sont supprimés de la file d'attente de réception et que les données correspondantes sont appliquées. Cette unité d'exécution est utilisée lorsque le paramètre **prune_method** est défini sur une valeur de 2 (valeur par défaut).

Détention de verrou (DB sur toutes les plateformes sauf z/OS)

Empêche les programmes Q Apply ayant le même schéma d'être exécutés sur un serveur.

Les unités d'exécution de navigation, d'agent et d'administration sont en général en cours de fonctionnement. Si les unités d'exécution agent sont en cours de fonctionnement mais que les données ne sont pas appliquées, vous pouvez utiliser les fenêtres Tableau de bord de réplication Q et Messages Q Apply, ou la table IBMQREP_APPLYTRACE pour rechercher des messages susceptibles d'expliquer pourquoi les données ne sont pas appliquées.

moniteur d'alertes de réplication

Un moniteur d'alertes de réplication possède les unités d'exécution suivantes :

Agent Surveille les tables de contrôle pour déterminer si des critères d'alerte définis par l'utilisateur ont été rencontrés et envoie des notifications d'alerte.

Administrateur

Gère les messages internes et les messages d'erreur.

Sérialisation

Gère les signaux entre les unités d'exécution du programme de surveillance.

Les unités d'exécution agent et d'administration généralement opérationnelles ou au repos. Les unités d'exécution de sérialisation sont en général en attente. Si l'unité d'exécution agent est opérationnelle mais que vous n'avez pas reçu les notifications d'alerte attendues, vous pouvez utiliser les tables de contrôle IBMSNAP_MONTRACE et IBMSNAP_MONTRAIL afin de rechercher des messages et d'autres informations susceptibles d'expliquer les raisons pour lesquelles les alertes n'ont pas été envoyées.

Lorsque vous vérifiez le statut des programmes Q Capture, Q Apply et du moniteur d'alertes de répliquions, si les messages d'indiquent pas les états habituels des unités d'exécution, il peut être nécessaire d'effectuer la procédure décrite dans le tableau 26.

Tableau 26. Descriptions des états des unités d'exécution et actions préconisées

Etat de l'unité d'exécution	Description	Action préconisée
En cours de fonctionnement	L'unité d'exécution a un traitement actif.	Aucune action.
Existe	L'unité d'exécution existe mais ne peut pas démarrer.	Si l'unité d'exécution ne sort pas de cet état, redémarrez le programme.
Démarrée	L'unité d'exécution est démarrée mais ne peut pas s'initialiser.	Recherchez d'éventuels problèmes de ressources, comme de trop nombreuses unités d'exécution ou un manque de puissance de traitement.
Initialisation	L'unité d'exécution est initialisée mais ne peut pas fonctionner.	Si l'unité d'exécution ne sort pas de cet état, redémarrez le programme.
Au repos	L'unité d'exécution est en veille et en sortira lorsqu'elle aura une tâche à effectuer.	Aucune action.
Arrêtée	L'unité d'exécution ne fonctionne pas.	Utilisez la fenêtre Messages Q Capture du Centre de répliquion ou la fenêtre Messages du moniteur du Centre de répliquion qui explique pourquoi l'unité d'exécution s'est arrêtée. Vous pouvez également rechercher des messages semblables dans les tables de contrôle IBMQREP_CAPTRACE, IBMQREP_APPLYTRACE ou IBMSNAP_MONTRACE.

Temps d'attente

Le temps d'attente est une mesure du temps nécessaire à la réplication des transactions entre le serveur Q Capture et le serveur Q Apply. Les programmes Q Capture et Q Apply enregistrent les données de performances qui vous permettent de suivre le temps d'attente entre les différentes étapes du processus de réplication. Ces statistiques vous aident à identifier les problèmes et à ajuster votre environnement.

La fenêtre Temps d'attente du centre de réplication vous permet de voir le temps nécessaire aux transactions pour passer du journal de récupération DB2 à la file d'attente d'envoi, de la file d'attente d'envoi à la file d'attente de réception et de la file d'attente de réception à la table cible. La fenêtre Temps d'attente de Q Capture offre une mesure approximative de la façon dont le programme Q Capture suit les changements apportés au journal.

Recommandation : Toute mesure du temps d'attente qui implique les transactions dupliquées entre les serveurs Q Capture et Q Apply distants peut être touchée par une différence d'heure entre les systèmes source et cible. Pour obtenir une mesure exacte, veillez à synchroniser les horloges.

Les sections suivantes apportent de plus amples informations sur chaque mesure du temps d'attente :

Temps d'attente de Q Capture

Le temps d'attente de Q Capture est la différence entre un moment donné dans le temps et l'horodatage de la dernière transaction validée. Cette mesure utilise les valeurs des colonnes MONITOR_TIME et CURRENT_LOG_TIME de la table de contrôle IBMQREP_CAPMON. En examinant ces mesures de temps d'attente en agrégat, vous pouvez voir comment le programme Q Capture parvient à suivre le journal de base de données.

Par exemple, si un programme Q Capture a inséré une ligne de données de performances dans le tableau IBMQREP_CAPMON (MONITOR_TIME) à 10:00 et que l'horodatage de la dernière transaction validée (CURRENT_LOG_TIME) indique 09:59, cela signifie que le temps d'attente de Q Capture est de une minute. Pour améliorer les performances de lecture des journaux, pensez à créer un schéma Q Capture supplémentaire et à déplacer certains des abonnements Q ou des publications vers le nouveau schéma. Chaque schéma supplémentaire ajoute une unité d'exécution agent pour lire le journal.

Temps d'attente de transaction Q Capture

Le temps d'attente de transaction Q Capture est le temps qui s'écoule entre la lecture, par le programme Q Capture, de la déclaration de validation pour une transaction dans le journal de récupération de DB2 et le placement dans une file d'attente d'envoi du message contenant la transaction. Cette statistique vous donne une idée du temps nécessaire au programme Q Capture pour rassembler les transactions en mémoire, filtrer les lignes et les colonnes en fonction des paramètres de l'abonnement Q ou de la publication et mettre les messages de transaction dans une file d'attente. Pour réduire le temps d'attente de transaction de Q Capture, envisagez les ajustements suivants :

- Augmenter la valeur du paramètre **memory_limit** qui détermine la quantité totale de mémoire attribuée par le programme Q Capture.

- Augmenter la valeur du paramètre **max_message_size**, défini lorsque vous créez une mappe de files d'attente pour la réplication ou la publication. Ce paramètre définit la quantité de mémoire qu'un programme Q Capture attribue à la génération de messages de transaction pour chaque file d'attente d'envoi. Si la taille maximale de message est trop petite, le programme Q Capture divise les transactions en plusieurs messages qui requièrent un temps de traitement supérieur et augmentent le temps d'attente.

Temps d'attente de file d'attente

Le temps d'attente de file d'attente mesure le temps qui s'écoule entre le moment où le programme Q Capture place une transaction dans une file d'attente d'envoi et le moment où le programme Q Apply reçoit la transaction de la file d'attente de réception. Cette statistique peut vous renseigner sur les performances de WebSphere MQ, bien que les performances du réseau soit également un facteur à prendre en compte dans les environnements répartis. De plus, un intervalle de validation plus long amène le programme Q Capture à attendre avant de valider certaines transactions qui ont été placées dans une file d'attente d'envoi. Ce retard peut contribuer au temps d'attente de file d'attente.

Temps d'attente Q Apply

Le temps d'attente du programme Q Apply mesure le temps qu'il faut à une transaction pour être appliquée à une table cible après que le programme Q Apply ait reçu la transaction d'une file d'attente de réception. Plus vous définissez d'unités d'exécution agent pour une file d'attente de réception, plus ce nombre sera petit. N'oubliez pas toutefois que le programme Q Apply retarde l'application d'une transaction impliquant des tables dépendantes jusqu'à ce que toutes les transactions antérieures dont elle dépend soient appliquées. Ce retard peut augmenter le temps d'attente de Q Apply.

Latence de bout en bout

La latence de bout en bout mesure le temps qui s'écoule entre la lecture, par le programme Q Capture, de l'enregistrement de journal pour une instruction de validation de transaction et la validation, par Q Apply, de la transaction sur la cible. Cette statistique est une mesure globale du temps d'attente de réplication Q.

Pour ouvrir la fenêtre Temps d'attente, cliquez avec le bouton droit de la souris sur le serveur Q Apply qui contient le programme Q Apply dont vous souhaitez afficher les statistiques puis sélectionnez **Rapports > Temps d'attente**.

Pour ouvrir la fenêtre Temps d'attente de Q Capture, cliquez avec le bouton droit de la souris sur le serveur Q Capture qui contient le programme Q Capture dont vous souhaitez afficher les statistiques puis sélectionnez **Rapports > Temps d'attente de Q Capture**.

Exceptions

Les exceptions sont des lignes que le programme Q Apply n'a pas appliquées aux cibles ou a retravaillées en raison de conflits ou d'erreurs SQL. Le programme Q Apply enregistre ces lignes, ainsi que les détails relatifs au conflit ou à l'erreur, dans la table `IBMQREP_EXCEPTIONS`.

Les conflits peuvent survenir lorsqu'une application autre que le programme Q Apply met à jour la cible. Par exemple, le programme Q Apply peut tenter

d'insérer une ligne qui existe déjà dans la cible. Si l'action de conflit de l'abonnement Q est F (forcer), Q Apply transforme l'instruction INSERT en UPDATE et l'instruction INSERT d'origine est consignée dans la table IBMQREP_EXCEPTIONS.

Les erreurs SQL couvrent un éventail de problèmes potentiels. Par exemple, si le journal de récupération du serveur cible est rempli, des erreurs SQL sont générées lorsque le programme Q Apply tente d'appliquer les lignes.

La fenêtre Exceptions du centre de réplication vous permet de consulter les détails relatifs aux lignes en erreur. Vous pouvez également utiliser les données enregistrées par le programme Q Apply dans sa table de contrôle pour récupérer les lignes qui n'ont pas été appliquées et pour en savoir plus sur les problèmes potentiels au sein de votre environnement de réplication.

La liste suivante indique les types de problèmes qui peuvent empêcher le programme Q Apply d'appliquer les lignes :

Erreurs SQL imprévues

Les lignes à l'origine des erreurs SQL n'étaient pas définies comme acceptables pour l'abonnement Q.

Erreurs SQL acceptables

Les lignes à l'origine des erreurs SQL ont été définies comme acceptables. Vous définissez les erreurs SQL acceptables lorsque vous créez un abonnement Q.

Conflits basés sur la valeur

Les lignes à l'origine des conflits tels que les tentatives de suppression ou de mise à jour d'une ligne qui n'existait pas dans la cible ou d'insertion d'une ligne qui existait déjà. Le programme Q Apply enregistre le type de détection de conflit utilisé (vérification des valeurs clé seulement, vérification des colonnes non clé et clé modifiées, vérification de toutes les colonnes clé et non-clé) et si la ligne a été appliquée malgré l'enregistrement. Lorsque vous créez un abonnement Q, vous pouvez préciser que le programme Q Apply doit appliquer certaines lignes même lorsqu'elles entraînent des conflits.

Conflits basés sur la version

Lignes à l'origine des conflits dans la réplication entre homologues, comme la tentative de mise à jour d'une ligne avec des valeurs d'une ligne plus ancienne ou la tentative d'insertion d'une ligne alors que la cible contient déjà une ligne plus récente avec la même clé.

Lorsque Q Apply enregistre des lignes dans la table IBMQREP_EXCEPTIONS, il enregistre également l'heure à laquelle l'erreur ou le conflit s'est produit, le motif de l'erreur et le type d'opération SQL ayant causé le problème. Le programme Q Apply enregistre également les noms de la file d'attente de réception et de l'abonnement Q et les informations de validation de la source pour la transaction.

Pour les erreurs SQL imprévues et pour les erreurs SQL acceptables, le programme Q Apply enregistre le code SQL et le code d'état SQL renvoyé par DB2 pour la transaction, ainsi que les marqueurs de message d'erreur de la structure SQLCA utilisée pour l'exécution de la transaction. Pour les conflits basés sur la valeur ou la version, le programme Q Apply enregistre les raisons pour lesquelles la ligne n'a pas pu être appliquée. Les lignes non appliquées sont sauvegardées au format SQL, dans la colonne TEXT de la table IBMQREP_EXCEPTIONS.

Dans la réplication entre homologues, le programme Q Apply enregistre les lignes qu'il n'a pas appliquées au serveur où s'est produit le conflit ou l'erreur SQL.

Pour ouvrir la fenêtre Exceptions, cliquez avec le bouton droit de la souris sur le serveur Q Apply qui contient le programme Q Apply dont vous souhaitez afficher les exceptions, puis sélectionnez **Rapports > Exceptions**.

Chapitre 20. Gestion d'un environnement de réplication Q et de publication d'événements

La réplication Q et la publication d'événements fonctionnent avec votre système de base de données et n'exigent que des changements mineurs de vos activités de base de données existantes. Gérer les systèmes source, les tables de contrôle et les tables cible peut permettre de garantir que votre système continue de fonctionner sans incident.

Considérations relatives à la gestion des systèmes source de réplication Q et de publication d'événements

Vous devez tenir compte de la disponibilité des tables source et des données de journal dans la réplication Q et la publication d'événements, afin que les programmes Q Capture et Q Apply soient toujours à même de fonctionner.

Le système source de réplication contient les objets suivants :

- Les tables source à partir desquelles vous souhaitez répliquer ou publier des données.
- Les données de journal que le programme Q Capture lit pour enregistrer les changements apportés aux tables source.

Gestion des tables source dans un environnement de réplication Q et de publication d'événements

Les sources de réplication sont des tables de base de données. Les tables source pour la réplication Q et la publication d'événements requièrent la même maintenance que les autres tables de base de données de votre système.

La réplication Q et la publication d'événements ne requièrent pas un accès direct aux tables source durant la plupart des tâches de traitement. Toutefois, ils doivent accéder à vos tables source directement dans les cas suivants :

- Le programme Q Apply exécute un chargement.
- Le programme Q Capture enregistre des données d'objet LOB.

Pour gérer des tables source pour les environnements de réplication Q et de publication d'événements :

- Poursuivez l'exécution des utilitaires et des routines de maintenance existants sur ces tables.
- Assurez-vous de disposer de droits d'accès en lecture pour vos tables source afin d'éviter l'interruption de l'exécution du programme Q Apply pendant un chargement.

Conservation de fichiers journaux pour la réplication Q et la publication d'événements

Si vous devez identifier les fichiers journaux requis par le programme Q Capture, vous pouvez utiliser une commande Q Capture sous Linux, UNIX et Windows, ou rechercher manuellement le journal requis le plus ancien en lisant une table de contrôle Q Capture et en faisant appel à un utilitaire DB2 sous z/OS.

Raisons pour lesquelles vous devez conserver les données de journal pour la réplication Q et la publication d'événements

Il est essentiel de conserver les données du journal pour la reprise de DB2 et la réplication Q ou la publication d'événements. De plus, vous devez vous assurer que les programmes Q Capture et DB2 sont entièrement exécutés avec un ensemble de journaux avant de procéder à leur suppression.

Vos journaux de reprise DB2 :

- Offrent des fonctions de reprise DB2
- Fournissent des informations à vos programmes Q Capture en cours d'exécution

Les données de journal résident dans les mémoires tampon de journal, les journaux actifs ou les journaux archivés. Chaque fois que le programme Q Capture démarre à chaud, il requiert tous les journaux DB2 créés depuis sont arrêt et tous les journaux DB2 qu'il n'a pas complètement traités.

Si vous exécutez le programme Q Capture de façon continue, il est généralement en phase avec le journal de reprise DB2. Si vous conservez également les fichiers journaux pendant au moins une semaine, vous pouvez continuer d'appliquer les procédures de conservation de journaux existantes. Cependant, vous devez changer de procédure de conservation de journaux en fonction de votre environnement de réplication Q et de publication d'événements si :

- Vous supprimez généralement les enregistrements de journaux dès que DB2 termine une sauvegarde et que ces enregistrements ne sont plus nécessaires pour la reprise aval.
- Vous êtes soumis à des contraintes de stockage et devez supprimer fréquemment vos journaux de reprise archivés.
- Vous exécutez le programme Q Capture occasionnellement et non de manière continue.

Détermination du plus ancien fichier journal requis par Q Capture (z/OS)

Vous pouvez lire une table de contrôle Q Capture et faire appel à un utilitaire DB2 pour déterminer le numéro d'ordre de journal (LSN) de l'enregistrement de journal le plus ancien dont le programme Q Capture a besoin sur les systèmes d'exploitation z/OS.

A propos de cette tâche

Utilisez DB2 pour référencer le numéro d'ordre du journal avec le fichier journal qui contient l'enregistrement le plus ancien. Le programme Q Capture a besoin de ce fichier journal et d'autres plus récents.

Procédure

Pour déterminer le fichier journal le plus ancien requis par le programme Q Capture :

1. Exécutez l'instruction SQL suivante pour obtenir le numéro d'ordre de journal de la transaction la plus récente que le programme Q Capture a consultée, traitée et enregistrée des ses tables de contrôle :

```
SELECT max(RESTART_SEQ)
FROM schéma.IBMQREP_CAPMON
WITH UR;
```

Voici un exemple de numéro d'ordre de journal :

0000555551F031230000

Ignorez les quatre premiers caractères de ce numéro, qui sont toujours 0000. Les 12 caractères suivants correspondent au numéro d'ordre de journal actif. (Cette valeur de 12 caractères est l'adresse relative en octets dans les environnements sans partage de données et correspond au numéro d'ordre d'enregistrement de journal (LRSN) dans les environnements avec partage de données.) Les quatre derniers caractères sont 0000 dans les environnements sans partage de données ; ils correspondent à l'ID de membre dans les environnements de partage de données.

2. Servez-vous de l'utilitaire DSNJU004 pour appeler l'utilitaire de la mappe de journal d'impression. Cet utilitaire affiche des informations sur les fichiers d'amorçage (BSDS). Par exemple :

```
# ACTIVE LOG COPY 1 DATA SETS
# START RBA/TIME END RBA/TIME DATE LTIME DATA SET INFORMATION
#-----
# 555551F03000 555551F05FFF 1998.321 12:48 DSN=DSNC710.LOGCOPY1.DS02
# 2001.57 15:46:32.2 2001.057 15:47:03.9 PASSWORD=(NULL) STATUS=TRUNCATED,REUSABLE
# 555551F06000 555551F09FFF 1998.321 12:49 DSN=DSNC710.LOGCOPY1.DS03
# 2001.57 15:47:32.2 2001.057 15:48:12.9 PASSWORD=(NULL) STATUS=TRUNCATED,REUSABLE
```

3. Comparez le numéro de journal actif à 12 caractères de la valeur RESTART_SEQ à l'adresse relative en octets de départ et à la plage correspondante d'adresses relatives en octets de fin dans chaque ligne affichée.
4. Recherchez la ligne pour laquelle le numéro de journal actif à 12 caractères de la table IBMQREP_CAPMON tombe entre l'adresse relative en octets de départ et l'adresse relative en octets de fin. Dans l'exemple :

```
# 555551F03000 555551F05FFF 1998.321 12:48 DSN=DSNC710.LOGCOPY1.DS02
# 2001.57 15:46:32.2 2001.057 15:47:03.9 PASSWORD=(NULL)STATUS=TRUNCATED,REUSABLE
```

5. Notez les informations correspondantes de l'Ensemble de données pour ce numéro de journal actif. Dans l'exemple :

```
DSNC710.LOGCOPY1.DS02
```

6. Notez la date et l'heure de ce fichier. Le programme Q Capture a besoin de ce fichier et d'autres plus récents pour redémarrer.

Détermination du plus ancien fichier journal requis par Q Capture (Linux, UNIX, Windows)

Vous pouvez utiliser une commande pour déterminer le fichier journal le plus ancien requis par le programme Q Capture et les journaux que vous pouvez supprimer sans risque sous les systèmes d'exploitation Linux, UNIX et Windows.

Avant de commencer

Pour émettre cette commande, le programme Q Capture doit être en cours d'exécution.

A propos de cette tâche

Cette commande utilise l'utilitaire db2flsn pour déterminer le fichier journal DB2 le plus ancien requis. Le programme Q Capture a besoin de ce journal, et d'autres plus récents, pour effectuer un redémarrage à un moment quelconque. Vous devez donc les conserver pour permettre une opération continue des programmes Q Capture. Vous pouvez supprimer les journaux plus anciens.

Procédure

Pour déterminer le plus ancien fichier journal requis par le programme Q Capture, procédez comme suit :

Utilisez la commande **asnqccmd** avec les paramètres suivants alors que le programme Q Capture est en cours d'exécution :

```
asnqccmd capture_server=nom_serveur capture_schema=schéma status show details
```

La commande retourne un rapport sur le statut du programme Q Capture, avec les informations suivantes :

- Chemin d'accès aux fichiers journaux DB2
- Plus ancien journal DB2 requis pour le redémarrage de Q Capture
- Fichier journal DB2 actuel détecté

Voici un exemple de sortie de ces informations :

```
Path to DB2 log files (DB2LOG_PATH) = /home2/szp/szp/  
NODE0000/SQL00002/SQLLOGDIR/  
Oldest DB2 log file needed for Q Capture restart (OLDEST_DB2LOG) = S0000043.LOG  
Current DB2 log file captured (CURRENT_DB2LOG) = S0000046.LOG
```

Recommandation : Lancez le programme Q Capture alors que DB2 est en service. Ceci permettra au programme Q Capture une lecture aussi proche que possible de la fin du journal DB2 (où les enregistrements les plus récents sont situés). La lecture à cet emplacement minimise le nombre d'anciens journaux requis par le programme Q Capture.

Considérations pour la gestion des dictionnaires de compression dans la réplication Q et la publication d'événements (z/OS)

Si vous utilisez les dictionnaires de compression DB2, vous devez coordonner le recours aux utilitaires avec les programmes Q Capture.

Mise à jour des dictionnaires de compression DB2

Lorsque le programme Q Capture demande des enregistrements de journal, DB2 doit décompresser les enregistrements de journal des tables stockées dans un espace table compressé. Pour la décompression, DB2 utilise le dictionnaire de compression en cours. Dans certains cas, il est possible que le dictionnaire de compression ne soit pas disponible. Le programme Q Capture prend des mesures différentes dans chaque cas :

Si le dictionnaire de compression est provisoirement indisponible

DB2 renvoie une erreur au programme Q Capture. Le programme Capture fait plusieurs tentatives pour continuer le traitement. Si le dictionnaire reste indisponible, le programme Q Capture émet un message ASN0011E et prend fin.

Si le dictionnaire de décompression est indisponible de manière définitive

Il est possible qu'un dictionnaire de compression soit perdu si vous faites appel à l'utilitaire REORG sans indiquer `KEEPDICTIONARY=YES`. Dans ce cas, le programme Q Capture émet un message ASN0011E, désactive l'abonnement Q et prend fin.

Avec APAR PK19539 (DB2 pour z/OS Version 8), DB2 conservera en mémoire une sauvegarde du dictionnaire de compression lorsque vous

utilisez l'utilitaire REORG sans indiquer KEEPDICTIONARY=YES. C'est pourquoi il est inutile de spécifier KEEPDICTIONARY=YES sauf si :

- Vous redémarrez DB2.
- Vous faites appel à l'utilitaire REORG à deux reprises pour le même espace table avant que le programme Q Capture ait lu tous les anciens enregistrements de journal pour cette table.

Pour éviter ces situations dans DB2 pour z/OS version 7, laissez le programme Q Capture traiter tous les enregistrements de journal pour une table avant d'effectuer une activité ayant un effet sur le dictionnaire de compression de cette table. Certaines des activités suivantes peuvent affecter les dictionnaires de compression :

- Altération d'un espace table afin de modifier son paramétrage de compression
- Utilisation de DSN1COPY pour copier des espaces table compressés d'un sous-système dans un autre, y compris à partir d'environnements de partage de données dans des environnements ne partageant pas les données
- Exécution de l'utilitaire REORG dans l'espace table

Verrouillage des dictionnaires de compression DB2

Vous devez également tenir compte de la disponibilité du dictionnaire de compression. Lorsque le programme Q Capture lit des enregistrements de journal compressés, DB2 verrouille l'espace table compressé source pour accéder au dictionnaire. Le programme Q Capture s'arrête si l'espace table compressé sur le système source se trouve dans l'état STOPPED lorsque l'interface de lecture du journal DB2 a besoin de ce verrou. À l'inverse, un utilitaire nécessitant un accès complet à l'espace table source ou exigeant que cet espace table soit dans l'état STOPPED peut être verrouillé par le verrou détenu par le programme Q Capture pendant qu'il lit le dictionnaire.

Pour éviter tout verrouillage temporaire dû à un verrou indisponible, interrompez le programme Q Capture lorsqu'un espace table compressé source doit être utilisé exclusivement par un utilitaire DB2 (ou tiers).

Gestion des tables de contrôle dans la réplication Q et la publication d'événements

Les tables de contrôle archivent des définitions d'objet et d'autres informations de contrôle spécifiques à la réplication. Bien que la taille de certaines tables de contrôle reste statique, d'autres tables de contrôle peuvent s'agrandir ou se réduire selon la taille de votre base de données et vos besoins de réplication.

Élagage des tables de contrôle dans la réplication Q et la publication d'événements

Certaines tables de contrôle de la réplication Q et la publication d'événements s'agrandissent de façon régulière. Les programmes de réplication élaguent la plupart de ces tables de contrôle, mais vous pouvez utiliser une commande système ou le Centre de réplication pour effectuer un élagage supplémentaire.

A propos de cette tâche

Le tableau 27 répertorie les tables de contrôle élaguées par le programme Q Capture qui s'agrandissent de manière régulière.

Tableau 27. Les tables de contrôle élaguées par le programme Q Capture

Table de contrôle	Paramètre spécifiant quelles lignes peuvent être élaguées
IBMQREP_CAPMON	monitor_limit
IBMQREP_CAPQMON	monitor_limit
IBMQREP_CAPTRACE	trace_limit
IBMQREP_SIGNAL	signal_limit

Le paramètre **prune_interval** définit la fréquence selon laquelle le programme Q Capture vérifie que les lignes peuvent être élaguées ou non.

Le tableau 28 répertorie les tables de contrôle élaguées par le programme Q Apply qui peuvent s'agrandir de manière régulière.

Tableau 28. Les tables de contrôle élaguées par le programme Q Apply

Table de contrôle	Paramètre spécifiant quelles lignes peuvent être élaguées
IBMQREP_APPLYMON	monitor_limit
IBMQREP_APPLYTRACE	trace_limit

La table IBMQREP_EXCEPTIONS peut également s'agrandir de manière régulière. Utilisez SQL pour élaguer manuellement les lignes contenant les exceptions que vous avez déjà traitées.

Le paramètre **prune_interval** définit la fréquence selon laquelle le programme Q Apply vérifie si les lignes peuvent être élaguées ou non.

Le moniteur d'alertes de réplication élague la table IBMSNAP_ALERTS, qui peut grandir de manière régulière. Le paramètre **alert_prune_limit** définit la quantité de données conservées dans la table. Le taux de croissance dépend de votre configuration de réplication et de vos paramètres.

Procédure

Pour effectuer un élagage complémentaire des tables de contrôle :

Utilisez l'une des méthodes suivantes :

Méthode	Description
Commande du système	<p>Associez le paramètre prune à la commande MODIFY sous z/OS ou à l'une des commandes suivantes sous Linux, UNIX, Windows et les services système UNIX (USS) sous z/OS :</p> <p>Tables de contrôle Q Capture asnqcmd</p> <p>Tables de contrôle Q Apply asnqacmd</p> <p>Tables de contrôle du moniteur asnmcmd</p>

Méthode	Description
Centre de réplication	<p>Utilisez l'une des fenêtres suivantes :</p> <p>Tables de contrôle Q Capture Utilisez la fenêtre Elaguer les tables de contrôle Q Capture. Pour ouvrir cette fenêtre, cliquez avec le bouton droit de la souris sur un serveur Q Capture dans le panneau de contenu et sélectionnez Elaguer les tables de contrôle Q Capture.</p> <p>Tables de contrôle Q Apply Utilisez la fenêtre Elaguer les tables de contrôle Q Apply. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur un serveur Q Apply dans le panneau de contenu et sélectionnez Elaguer les tables de contrôle Q Apply.</p> <p>Tables de contrôle du moniteur Utilisez la fenêtre Elaguer les tables de contrôle du moniteur. Pour ouvrir cette fenêtre, cliquez avec le bouton droit de la souris sur le qualificatif du moniteur dans le panneau de contenu et sélectionnez Elaguer les tables de contrôle du moniteur.</p>

Considérations relatives à l'utilisation de l'utilitaire RUNSTATS sur les tables de contrôle dans un environnement de réplication Q et de publication d'événements

L'optimiseur peut améliorer l'accès à vos tables de contrôle de réplication Q et de publication d'événements.

L'utilitaire RUNSTATS met à jour les statistiques concernant les caractéristiques physiques de vos tables et index associés. Continuez d'exécuter l'utilitaire RUNSTATS sur vos tables existantes avec la même fréquence que lorsque vous utilisez la réplication Q. Toutefois, n'exécutez qu'une fois l'utilitaire RUNSTATS sur les tables de contrôle qui s'agrandissent de façon régulière (et qui sont régulièrement élaguées) lorsque ces tables contiennent des quantités de données importantes. RUNSTATS transmet les informations utiles concernant ces tables dynamiques lorsque celles-ci atteignent leur taille de niveau de production maximum et l'optimiseur obtient les statistiques nécessaires pour déterminer la meilleure stratégie pour accéder aux données.

Réorganisation des tables de contrôle

Réorganisez régulièrement toutes les tables de contrôle qui changent fréquemment de taille afin d'éliminer les données fragmentées et de récupérer de l'espace.

A propos de cette tâche

Réorganisez les tables de contrôle suivantes une fois par semaine :

- IBMQREP_APPLYMON
- IBMQREP_CAPMON
- IBMQREP_CAPQMON
- IBMQREP_APPLYTRACE
- IBMQREP_CAPTRACE
- IBMSNAP_MONTRAIL

- IBMSNAP_MONTRACE
- IBMQREP_SIGNAL

En fonction de votre environnement de réplication et vos configurations, il se peut que vous deviez réorganiser les tables suivantes :

- IBMQREP_DELTOMB (configurations entre homologues)
- IBMQREP_DONEMSG
- IBMQREP_EXCEPTIONS
- IBMSNAP_ALERTS (moniteur d'alertes de réplication)

Remarque : Usez de prudence lors de la réorganisation de la table IBMQREP_SIGNAL alors que les programmes de réplication sont actifs. Si l'état d'un abonnement Q ou d'une publication est modifié alors que l'utilitaire REORG est en cours, un conflit au niveau de la table IBMQREP_SIGNAL peut poser problème.

Procédure

Pour réorganiser les tables de contrôle :

- **z/OS** Utilisez l'utilitaire REORG avec l'option PREFORMAT. L'option PREFORMAT accélère le traitement d'insertion du programme Q Capture.
- **Linux UNIX Windows** Utilisez la commande **REORG**.

Quand les programmes de réplication ne peuvent pas se connecter à leur serveur DB2

Pour une exécution correcte, les programmes Q Capture, Q Apply et le moniteur d'alertes de réplication doivent pouvoir se connecter au serveur DB2 contenant leurs tables de contrôle. Lorsqu'un programme de réplication ne peut pas accéder à ses tables de contrôle, il émet un message d'erreur approprié et s'arrête.

Les problèmes de connectivité nécessitent généralement que vous redémarriez le programme lorsque la connectivité revient. Par exemple, si un programme Q Apply s'arrête parce que le serveur DB2 contenant ses tables de contrôle a été arrêté ou mis au repos, il suffit de redémarrer le programme Q Apply lorsque le serveur DB2 est en cours d'exécution.

Si le programme peut se connecter au serveur DB2 mais reçoit une erreur SQL lorsque le programme essaie d'accéder à ses tables de contrôle, effectuez l'intervention appropriée pour cette erreur SQL, puis redémarrez le programme. Par exemple, si l'erreur SQL indique qu'une table de contrôle doit être récupérée, utilisez une procédure de reprise standard DB2 pour récupérer la table, puis redémarrez le programme.

Gestion des tables cible

La gestion des tables cible est semblable à la gestion d'autres tables. Toutefois, vous devez prendre en compte les opérations du programme Q Apply dans votre routine de maintenance pour ces tables.

A propos de cette tâche

Lorsque vous avez besoin d'effectuer des opérations de maintenance sur une table cible comme la réorganisation d'une table, il est nécessaire d'empêcher les applications comme le programme Q Apply d'utiliser la table jusqu'à ce que les opérations de maintenance soient terminées. Il est possible que le programme Q Apply s'arrête en raison d'une erreur s'il essaie de mettre une table à jour pendant des opérations de maintenance.

Différentes options vous permettent d'effectuer des opérations de maintenance sur les tables cible en fonction de vos exigences. Si vous avez besoin de réorganiser une seule table à un moment donné, vous pouvez définir l'abonnement Q pour cette table de façon à utiliser le mode de déversement temporaire. Votre configuration et vos exigences détermineront quelle option choisir.

Procédure

Pour gérer des tables cible :

- Gérez les tables sur le serveur cible de la même manière que pour les autres tables de votre système de base de données.
- Exécutez les routines de sauvegarde et de maintenance actuelles sur ces tables cible, que ces dernières soient des tables de base de données existantes ou des tables configurées pour être générées automatiquement par la réplication Q.
- Choisissez l'option correspondant le mieux à vos exigences. Certaines méthodes affectent plus de tables que d'autres méthodes.

Option	Description
<p>Placez un abonnement Q en mode de déversement</p>	<p>Cette option place un abonnement Q pour une table cible unique sur l'état de déversement. Le programme Q Apply conserve les modifications provenant de la table source dans une file d'attente auxiliaire temporaire. Alors que les modifications sont déversées, vous pouvez effectuer les opérations de maintenance sur la table cible. La file d'attente auxiliaire est créée en fonction de la définition de votre file d'attente modèle. Assurez-vous que la longueur maximale de votre file d'attente modèle est suffisamment grande pour conserver les lignes déversées.</p> <p>Utilisez les paramètres spillsub et resumesub pour passer l'abonnement Q à l'état déversement, puis reprendre les opérations normales ensuite. Vous pouvez utiliser ces paramètres avec la commande MODIFY sous z/OS ou avec la commande asnqacmd sous Linux, UNIX, Windows et UNIX System Services sous z/OS. L'exemple suivant utilise la commande MODIFY :</p> <pre>f myqapp,spillsub=nom_file_réception:nom_abonnement_q</pre> <p>Restriction : Si vos tables ont des contraintes d'intégrité référentielle, vous devez utiliser une autre méthode comme l'arrêt du traitement des messages sur la file d'attente de réception.</p>
<p>Arrêt du traitement des messages dans une file d'attente de réception</p>	<p>Vous pouvez utiliser le paramètre stopq avec la commande MODIFY sous z/OS ou avec la commande asnqacmd sous Linux, UNIX, Windows et UNIX System Services on z/OS, pour demander au programme Q Apply d'arrêter le traitement des messages pour la file d'attente de réception. La file d'attente de réception continue de recevoir des transactions, qui sont conservées jusqu'à ce que la file d'attente de réception soit réactivée.</p> <p>Cette méthode affecte toutes les tables qui utilisent la file d'attente de réception. Si vous devez gérer des tables qui partagent une file d'attente de réception ou si vos tables possèdent des contraintes d'intégrité référentielle, cette méthode est recommandée.</p> <p>Utilisez le paramètre stopq pour arrêter le traitement des messages, et le paramètre startq pour reprendre le traitement des messages une fois la maintenance terminée.</p>

Option	Description
Arrêter le programme Q Apply	Il est possible que votre configuration nécessite l'arrêt du programme Q Apply pour pouvoir gérer vos tables cible. Cela affecte l'ensemble des tables cible du serveur Q Apply. Les messages des abonnements Q sont placés dans des files d'attente de réception et seront traités lorsque le programme Q Apply sera à nouveau démarré.

Remarques relatives à la redéfinition de modules et de plans pour la réplication Q et la publication d'événements

Les modules pour la réplication Q et la publication d'événements doivent être définis avec des lectures non validées d'isolement afin de maintenir les performances du système à un niveau optimal.

De nombreux modules et plans pour la réplication Q et la publication d'événements sont redéfinis avec des lectures non validées d'isolement. Les programmes de maintenance internes que vous utilisez pour la redéfinition automatique de ces modules et de ces plans peuvent provoquer des incidents entre le programme Q Capture et le programme Q Apply. Si vos programmes de maintenance internes redéfinissent les modules de réplication avec des options standard comme la lecture non reproductible, cela génère des interférences avec les programmes Q Capture et Q Apply.

z/OS

Les accès aux programmes Q Capture, Q Apply et aux modules Common sont définis automatiquement. Vous pouvez utiliser l'échantillon ASNQBNDL z/OS pour redéfinir les modules ASNCOMMON, ASNQCAPTURE, ASNQAPPLY et ASNMON dans un sous-système DB2.

La réplication Q et la publication d'événements définissent le paramètre VERSION AUTO lorsque les modules sont précompilés. DB2 for z/OS libère automatiquement tout module antérieur aux deux versions les plus récentes.

Linux UNIX Windows

Les modules pour le programme Q Capture, le programme Q Apply et le moniteur d'alertes de réplication sont définis automatiquement lors de la première connexion du programme à ses tables de contrôle.

Chapitre 21. Comparaison et réparation de tables

Les commandes **asntdiff** et **asntrep** détectent et réparent les différences entre les tables. Dans la réplification Q et la réplification SQL, les commandes permettent d'identifier rapidement les différences et de synchroniser les tables sans effectuer de régénération intégrale ou de chargement de la table cible.

A propos de cette tâche

Les tables source et cible peuvent perdre la synchronisation, par exemple, si une table cible est modifiée de manière inattendue par un utilisateur ou une application, ou en cas d'indisponibilité du réseau étendu ou du système cible.

Les commandes **asntdiff** et **asntrep** sont exécutées indépendamment des programmes Q Capture, Q Apply, Capture et Apply. Elles utilisent DB2 SQL pour extraire des données de la table source et de la table cible, et n'utilisent pas les files d'attente WebSphere MQ. Les utilitaires de comparaison et de réparation ne dépendent pas des journaux, des déclencheurs ou du niveau d'isolement.

Procédure

Pour comparer et réparer des tables, exécutez la commande **asntdiff** puis la commande **asntrep**.

Utilitaire de comparaison de tables (asntdiff)

La commande **asntdiff** compare les colonnes d'une table aux colonnes correspondantes d'une autre table et génère une liste des différences entre les deux tables sous la forme d'une table DB2.

Pour utiliser l'utilitaire de comparaison, exécutez la commande **asntdiff** et indiquez le nom d'un abonnement Q (réplification Q) ou d'un membre d'ensemble d'abonnements (réplification SQL) qui contient les tables source et cible que vous souhaitez comparer. Vous pouvez également utiliser des instructions SQL dans un fichier en entrée pour préciser les tables à comparer.

Les sections suivantes expliquent comment utiliser la commande **asntdiff** :

- «Présentation de la commande **asntdiff**», à la page 342
- «Quand utiliser l'utilitaire de comparaison», à la page 342
- «Stockage des différences», à la page 343
- «Autorisations obligatoires», à la page 344
- «Restrictions concernant les colonnes clés dans la source et la cible», à la page 344
- «Considérations relatives au type de données», à la page 344
- «Effets du filtrage», à la page 345
- «Comparaisons basées sur les requêtes au lieu des abonnements», à la page 346
- «Comparaison d'un sous-ensemble de lignes de table», à la page 347

Présentation de la commande **asntdiff**

Vous pouvez exécuter la commande **asntdiff** sur les systèmes d'exploitation Linux, UNIX, Windows et z/OS. La commande compare les tables présentes sur les systèmes d'exploitation Linux, UNIX, Windows, z/OS ou System i. La commande **asntdiff** peut être utilisée avec des sources et des cibles fédérées si les colonnes correspondantes des deux tables contiennent le même type de données.

z/OS Le modèle de travail ASNTDIFF dans l'ensemble de données SASNSAMP fournit des informations complémentaires et spécifiques à la plateforme z/OS.

Pour la réplication Q, la cible doit être une table de copie utilisateur ou une table CCD (consistent-change-data) condensée et complète. Les cibles des procédures stockées ne sont pas prises en charge. Pour la réplication SQL, la réplique doit être une table utilisateur, une table des points de cohérence, une table réplique, une table de copie utilisateur ou une table CDD (consistent-change-data) condensée et complète.

Lorsque vous exécutez la commande, vous indiquez une clause WHERE SQL qui identifie de manière unique le membre de l'ensemble d'abonnements Q ou d'abonnements :

Réplication Q

La clause WHERE identifie une ligne de la table de contrôle IBMQREP_SUBS au niveau du serveur Q Capture en fonction de la valeur de la colonne SUBNAME. Par exemple :

```
where="subname = 'my_qsub' "
```

Réplication SQL

La clause WHERE identifie une ligne de la table IBMSNAP_SUBS_MEMBR au niveau du serveur de contrôle Apply en fonction de la valeur de la colonne SET_NAME. Par exemple :

```
où="set_name = 'my_set' et source_table='EMPLOYEE' "
```

Il sera peut-être nécessaire d'utiliser des prédicats supplémentaires dans la clause WHERE pour identifier de manière unique le membre de l'ensemble d'abonnements. Par exemple, vous devrez éventuellement ajouter la colonne APPLY_QUAL, SOURCE_OWNER, TARGET_OWNER ou TARGET_TABLE de la table IBMSNAP_SUBS_MEMBR à la clause.

Quand utiliser l'utilitaire de comparaison

L'utilitaire doit être exécuté de préférence lorsque les tables source et cible sont stables. Vous souhaiterez peut-être exécuter l'utilitaire lorsque les programmes Q Capture et Q Apply, ou les programmes Capture et Apply sont inactifs. Par exemple, vous pouvez exécuter l'utilitaire lorsque le programme Q Capture a atteint la fin du journal de récupération DB2 et que tous les changements sont validés dans la cible. Si les applications n'ont pas encore terminé la mise à jour de la source, la comparaison risque de ne pas être exacte.

Si les programmes de réplication sont en cours d'exécution, vous serez peut-être alors amené à exécuter la commande **asntdiff** plusieurs fois pour obtenir un panorama complet des différences changeantes entre les tables source et cible.

Stockage des différences

La commande **asntdiff** crée une table des différences dans la base de données ou le sous-système source pour y stocker les différences découvertes.

La table des différences s'appelle *schéma*.ASNTDIFF, où *schéma* correspond à la valeur indiquée dans le paramètre DIFF_SCHEMA. Si le schéma n'est pas indiqué, sa valeur par défaut est ASN. Vous pouvez également utiliser le paramètre DIFF pour indiquer un nom de table.

Par défaut, la table des différences est créée dans l'espace table d'utilisateur DB2 par défaut. Vous pouvez indiquer un espace table existant différent à l'aide du paramètre DIFF_TABLESPACE.

La table des différences contient deux ou plusieurs colonnes. Une colonne est nommée DIFF et contient un espace blanc à la fin sous Linux, UNIX et Windows. La valeur de la colonne DIFF est un caractère qui indique une opération d'insertion, de mise à jour ou de suppression, suivie d'une valeur numérique qui indique la table contenant une ligne avec les différences. Les autres colonnes contiennent la valeur des colonnes clés de réplication. La table des différences contient une ligne pour chaque ligne ne trouvant aucune correspondance dans la table cible.

La table des différences utilise trois identificateurs qui indiquent l'opération nécessaire pour modifier la table cible afin qu'elle corresponde à la table source :

D (delete)

Indique qu'une ligne contenant la valeur clé existe uniquement dans la cible et pas la source.

U (update)

Indique que des lignes contenant la même valeur clé existent dans la source et la cible, mais qu'au moins une colonne non clé est différente dans la cible.

I (insert)

Indique qu'une ligne contenant la valeur clé existe uniquement dans la source et pas la cible.

La valeur ? 1 indique qu'une ou plusieurs colonnes source contiennent un caractère non valide.

La valeur ? 2 indique qu'une ou plusieurs colonnes cible contiennent un caractère non valide.

Exemple :

La liste de valeurs suivante est renvoyée en comparant une table EMPLOYEE de la source avec une copie cible de la même table. La colonne clé pour la réplication est le nombre de l'employé, EMPNO :

```
DIFF  EMPNO
U 2    000010
I 2    000020
I 2    000040
D 2    000045
I 2    000050
D 2    000055
```

La première ligne de l'exemple montre qu'une ligne contenant la valeur clé 000010 existe dans les tables source et cible, mais au moins une colonne non clé de la cible contient une valeur différente. Les deux lignes suivantes montrent que les lignes contenant les valeurs clés 000020 et 000040 n'existent que dans la source. La quatrième source montre qu'une ligne contenant la valeur clé 000045 n'existe que dans la cible.

Les valeurs ? 1 et ? 2 ne sont pas illustrées dans l'exemple.

Autorisations obligatoires

Ces autorisations de base de données sont obligatoires pour l'utilitaire de comparaison :

- Droits d'accès aux tables en cours de comparaison et aux tables de contrôle de réplication, à moins que l'option -f (fichier) ne soit utilisée
- **Linux UNIX Windows** Privilèges de lecture pour le fichier de mot de passe si le mot clé PWDFILE est utilisé
- Privilège d'ECRITURE pour le répertoire spécifié par le mot clé DIFF_PATH
- Pour créer les tables de différences, il faut des droits d'accès CREATETAB sur la base de données source et le privilège d'UTILISATION sur l'espace de table. Par ailleurs, l'un des privilèges suivants est requis :
 - Droits d'accès IMPLICIT_SCHEMA sur la base de données, si le nom de schéma implicite ou explicite de la table n'existe pas
 - Privilège CREATEIN sur un schéma existant si la table est créée dans ce schéma

Sur z/OS, si l'ID utilisateur qui exécute asntdiff n'a pas les droits pour créer des tables, vous pouvez utiliser le mot clé SQLID pour spécifier un identificateur d'autorisation pouvant être utilisé pour créer la table de différences.

- Privilège DROPIN sur le schéma pour déplacer la table de différences, sauf si DIFF_DROP=N est utilisé
- Privilèges SELECT, DELETE et INSERT sur la table de différences (au niveau de la source). Le nom du schéma par défaut est ASN et le nom de la table par défaut est ASNTDIFF.

Restrictions concernant les colonnes clés dans la source et la cible

L'utilitaire asntdiff prend en charge les jeux de caractères à octets multiples lorsque la base de données est définie avec SYSTEM ou IDENTITY. Cependant, les colonnes qui sont utilisées en tant que clés pour la réplication dans les tables source et cible doivent utiliser des caractères à un octet pour que l'utilitaire compare les tables.

Dans une base de données Linux, UNIX ou Windows au format Unicode, le nombre de caractères des données clés ne peut pas être supérieur à celui du sous-ensemble de base ASCII américain (256 premiers caractères ASCII). Sinon, l'utilitaire asntdiff ne pourra pas comparer les tables.

Considérations relatives au type de données

Lors de l'utilisation de asntdiff, vous devez prendre en compte les types de données des tables que vous comparez.

Différents types de données dans les sources et les cibles

L'utilitaire de comparaison peut construire deux instructions SQL SELECT qui reposent sur la description d'un abonnement. Pour connaître les différences entre les tables source et cible, l'utilitaire compare les données qui résultent de l'exécution de deux instructions. Les types de données et les longueurs des colonnes pour les deux instructions SQL doivent être identiques.

Réplication SQL

L'utilitaire construit l'instruction SQL pour la source à l'aide de la colonne EXPRESSION de la table IBMSNAP_SUBS_COLS.

Réplication Q

Les types de données pour la source et la cible doivent être identiques.

Types de données non pris en charge

L'utilitaire de comparaison ne prend pas en charge les comparaisons entre les types de données suivants :

Colonnes non-clés

DECFLOAT, BLOB_FILE, CLOB_FILE, DBCLOB_FILE

Colonnes clés

DECFLOAT, BLOB, CLOB, DBCLOB, VARGRAPHIC, GRAPHIC, LONG_VARGRAPHIC, BLOB_FILE, CLOB_FILE, DBCLOB_FILE, XML

Comparaison des données de type GRAPHIC

Il est possible que les colonnes contenant des données de type GRAPHIC dans la source et la cible ne soient pas identiques lorsque vous faites appel à l'utilitaire pour comparer les tables source et cible. Les données GRAPHIC contenues dans les colonnes DB2 sont suivies de caractères de remplissage. Ces caractères peuvent prendre la forme d'espaces à un ou deux octets, selon la page de codes dans laquelle la base de données a été créée. Ces caractères de remplissage peuvent être à l'origine de la non-correspondance des données des tables source et cible, surtout si celles-ci sont associées à des pages de codes différentes. Ces caractères de remplissage s'appliquent uniquement aux types de données GRAPHIC et non aux autres types de données graphiques tels que VARGRAPHIC ou LONG VARGRAPHIC.

Pour comparer des colonnes contenant des données de type GRAPHIC, vous devez supprimer les caractères de remplissage dans les données avant de comparer les tables source et cible à l'aide de la fonction scalaire DB2 rtrim(<column>. Cette fonction élimine les différences de page de codes pour les espaces à un ou deux octets et veille à ce que l'utilitaire compare les données GRAPHIC de manière cohérente.

Restriction TIMESTAMP WITH TIMEZONE

L'utilitaire de comparaison ne prend pas en charge les comparaisons qui impliquent le type de données TIMESTAMP WITH TIMEZONE introduit dans DB2 pour z/OS version 10.

Effets du filtrage

Dans certains cas, les différences entre les tables source et cible sont volontaires, comme par exemple, si vous utilisez une condition de recherche dans la réplication

Q pour filtrer les lignes répliquées. L'utilitaire ne soulignera pas les différences entre les tables source et les tables cible qui résultent de prédicats ou de suppressions annulées.

Filtrage de lignes

L'utilitaire de comparaison utilise les informations des tables de contrôle de réplication pour éviter l'affichage des différences intentionnelles :

Réplication SQL

L'utilitaire s'appuie sur la colonne PREDICATES de la table IBMSNAP_SUBS_MEMBR pour sélectionner des lignes des tables source. La valeur de la colonne UOW_CD_PREDICATES est ignorée (asntdiff analyse directement la table source, alors que le programme Apply analyse la table de modification des données).

Réplication Q

L'utilitaire s'appuie sur la valeur de la colonne SEARCH_CONDITION de la table IBMQREP_SUBS pour construire la clause WHERE de l'instruction SELECT.

Opérations de suppression annulées

Dans la réplication Q, vous pouvez annuler la réplication des opérations de suppression de la table source. Si vous ne répliquez pas les opérations de suppression, les lignes qui existent dans la table cible n'existent peut-être pas dans la table source. Lorsque la valeur SUPPRESS_DELETES d'un abonnement Q est Y, l'utilitaire asntdiff ignore les lignes qui sont propres à la cible et ne signale aucune différence. Un avertissement indiquant le nombre de lignes supprimées s'affiche.

L'option asntdiff -f (fichier en entrée) ne prend pas en charge SUPPRESS_DELETES car il base la comparaison des tables sur une instruction SQL SELECT plutôt que sur la définition d'abonnement Q.

Comparaisons basées sur les requêtes au lieu des abonnements

L'option de commande **asntdiff -f** vous permet d'effectuer une différenciation en utilisant des instructions SQL SELECT qui sont lues à partir d'un fichier en entrée. Cette option offre une plus grande souplesse pour procéder à une différenciation entre deux tables génériques. L'option **asntdiff -f** n'utilise pas de définitions de réplication pour déterminer les tables et lignes à comparer comme le fait la commande asntdiff standard.

L'option **asntdiff -f** fonctionne pour toutes les tables sous Linux, UNIX, Windows et z/OS. Pour plus de détails sur cette option, voir «Option de commande asntdiff -f (fichier en entrée)», à la page 461.

En plus des instructions SELECT, le fichier en entrée contient les informations de base de données source et cible, les informations de différenciation entre les tables et des paramètres facultatifs qui spécifient des méthodes de traitement des différences. Vous pouvez utiliser un fichier de mots de passe créé par la commande **asnpwd** pour spécifier un ID utilisateur et un mot de passe afin de se connecter aux bases de données source et cible.

Remarque : Pour comparer les colonnes XML DB2 à l'aide de l'option **asntdiff -f**, vous devez sérialiser la colonne XML en tant que type de données d'objet CLOB en utilisant la fonction scalaire XMLSERIALIZE. Par exemple, cette instruction SELECT dans le fichier en entrée compare la colonne XMLColumn dans la table

source Table 1 à la même colonne dans une table d'une autre base de données (TARGET_SELECT utiliserait la même fonction) :

```
SOURCE_SELECT="select ID, XMLSERIALIZE(XMLColumn AS CLOB)
AS XMLColumn from Table1 order by 1"
```

Comparaison d'un sous-ensemble de lignes de table

Vous pouvez utiliser le paramètre RANGECOL de l'utilitaire **asntdiff** pour comparer seulement quelques lignes de deux tables. Ce paramètre indique une plage de lignes de la table source qui sont liées par deux horodatages. Vous indiquez le nom d'une colonne DATE, TIME ou TIMESTAMP dans la table source, puis utilisez l'une des trois clauses pour spécifier la plage. Lorsque vous comparez des tables impliquées dans la réplication entre homologues, vous pouvez utiliser la colonne IBMQREPVERTIME générée par IBM pour la colonne source dans la clause plage.

Le paramètre RANGECOL n'est pas valide pour l'option **asntdiff -f** (fichier en entrée). Vous pouvez utiliser une clause WHERE SQL dans le fichier en entrée pour obtenir des résultats similaires.

Exécution de l'utilitaire asntdiff en mode parallèle (sous z/OS)

En utilisant l'option PARALLEL=Y en combinaison avec la commande asntdiff, vous pouvez exécuter l'utilitaire de comparaison de tables dans un mode parallèle qui offre des performances optimales pour la comparaison de tables volumineuses.

En mode parallèle, l'utilitaire asntdiff utilise jusqu'à 21 unités d'exécution pour la comparaison des données de deux tables distinctes, ce qui réduit considérablement le temps de traitement tout en préservant la précision de la comparaison. Dans ce mode :

- L'utilitaire partitionne en interne les deux tables et compare ces partitions en parallèle.
- La récupération des lignes pour chaque paire de partitions est réalisée en parallèle.
- Les différences trouvées dans chaque paire de partitions sont ensuite combinées pour obtenir le résultat global.

Cette méthode réduit le temps de traitement et diminue l'utilisation de mémoire car elle ne nécessite pas de matérialiser des résultats intermédiaires volumineux. Le mode parallèle minimise par ailleurs le trafic réseau car les calculs du total de contrôle se font au niveau de chaque base de données.

Pour utiliser asntdiff en mode parallèle, il est recommandé (mais facultatif) d'avoir deux tables ayant des colonnes de date, d'heure ou d'horodatage et un index à entrées uniques ou clé primaire. Les deux tables doivent être sous DB2 pour le système d'exploitation z/OS et utiliser la même page de codes et la même séquence de classement.

Les rubriques suivantes détaillent davantage l'utilisation du mode parallèle.

- «Exigences d'installation», à la page 348
- «Autorisations obligatoires», à la page 348
- «Restrictions», à la page 349
- «Conseils d'utilisation», à la page 349

Exigences d'installation

Pour utiliser l'utilitaire asntdiff en mode parallèle, vous devez installer une procédure stockée (ASNTDSP) au niveau des systèmes contenant une table à comparer. Le modèle de travail ASNTDSP est compris dans le jeu de données SASNSAMP. Pour de meilleurs résultats, utilisez un environnement d'application avec NUMTCB = 8 - 15.

Le code suivant définit ASNTDSP :

```
CREER PROCEDURE ASN.TDIFF
( IN      SELECTSTMT      VARCHAR(32700),
  IN      GTTNAME         VARCHAR(128),
  IN      OPTIONS         VARCHAR(1331),
  OUT     RETURN_BLOCK_CRC VARCHAR(21),
  OUT     RETURN_NUM_ROWS  INTEGER,
  OUT     RETURN_CODE      INTEGER,
  OUT     RETURN_MESSAGE   VARCHAR(30000))
PARAMETER CCSID EBCDIC
EXTERNAL NAME ASNTDSP
LANGUAGE C
PARAMETER STYLE GENERAL WITH NULLS
COLLID ASNTDIFF
WLM ENVIRONMENT !!WLMENV4!!
MODIFIES SQL DATA
ASUTIME NO LIMIT
STAY RESIDENT YES
PROGRAM TYPE MAIN
SECURITY USER
RUN OPTIONS 'TRAP(OFF),STACK(,,ANY,),POSIX(ON) '
COMMIT ON RETURN NO;
```

Autorisations obligatoires

Ces autorisations et privilèges de base de données (ou droits plus élevés) sont nécessaires pour l'exécution de asntdiff en mode parallèle :

- SELECT sur les tables en cours de comparaison.
- Privilèges pour la table de différences :
 - CREATEIN et DROPIN sur un schéma existant si la table est créée dans ce schéma
 - Si le paramètre DIFF_TABLESPACE n'est pas spécifié, droits CREATETAB et CREATETS sur la base de données par défaut DSNDB04. Notez que sous DB2 pour z/OS version 10, si la clause IN n'est pas spécifiée avec CREATE TABLE, le privilège CREATETAB est obligatoire sur la base de données DSNDB04.
 - Si le paramètre DIFF_TABLESPACE est explicitement spécifié, le droit CREATETAB sur la base de données contenant les privilèges DIFF_TABLESPACE et USE sur l'espace de table spécifié par DIFF_TABLESPACE
- Pour créer les tables temporaires globales créées dans les bases de données des fichiers de travail, les privilèges CREATETMTAB au niveau de chaque base de données impliquée
- Privilèges SELECT, DELETE et INSERT sur les bases de données où la table des différences est créée. Le nom du schéma par défaut est ASN et le nom de la table par défaut est ASNTDIFF.
- Privilèges SELECT sur les tables du catalogue SYSIBM.SYSDUMMY1, SYSIBM.SYSTABLES, SYSIBM.SYSKEYS et SYSIBM.SYSINDEXES
- Privilèges EXECUTE sur la procédure ASN.TDIFF

- Privilèges EXECUTE sur le pack ASNTDIFF.ASNTDSP
- Privilèges EXECUTE sur le plan ASNRD101

Assurez-vous que les autorisations DB2 nécessaires soient accordées à l'ID utilisateur qui exécute asntdiff avant de démarrer l'utilitaire. Vous pouvez utiliser les paramètres TARGET_SQLID et SOURCE_SQLID pour modifier la valeur de CURRENT_SQLID pour un identificateur d'autorisation disposant de droits suffisants.

Restrictions

- Les deux tables en cours de comparaison doivent avoir la même page de codes et la même séquence de classement. Si ce n'est pas le cas, utilisez l'option PARALLEL=N (par défaut) pour la comparaison des tables.
- Lors d'une utilisation en mode parallèle, l'utilitaire asntdiff doit être exécuté depuis z/OS en tant que travail par lots qui utilise JCL.
- Les tables en cours de comparaison doivent résider sur z/OS.
- Seule l'option -F PARM est prise en charge lorsque asntdiff est exécuté en mode parallèle.
- Les instructions SELECT prises en charge que vous utilisez avec les paramètres SOURCE_SELECT et TARGET_SELECT doivent avoir la structure suivante :

```
SELECT xxx FROM yyy (WHERE zzz) ORDER BY aaa
```

La clause WHERE est facultative.

- Les types de données pris en charge pour les colonnes non-clés sont DATE, TIME, TIMESTAMP, VARCHAR, CHAR, LONG VARCHAR, FLOAT, REAL, DECIMAL, NUMERIC, BIGINT, INTEGER, SMALLINT, ROWID, VARBINARY, BINARY, VARGRAPH, GRAPHIC, LONGGRAPH.
- Les types de données pris en charge pour les colonnes clés sont DATE, TIME, TIMESTAMP, VARCHAR, CHAR, LONG VARCHAR, FLOAT, REAL, DECIMAL, NUMERIC, BIGINT, INTEGER, SMALLINT, ROWID, VARBINARY, BINARY.

Pour les autres restrictions, voir «Option de commande asntdiff -f (fichier en entrée)», à la page 461.

Conseils d'utilisation

- Les colonnes utilisées dans les clauses WHERE et ORDER BY doivent utiliser un index. Les colonnes que vous spécifiez dans la clause ORDER BY doivent suivre le même ordre de colonnes d'index et les mêmes attributs croissants/décroissants. Utilisez les utilitaires RUNSTATS et REORG pour maintenir à jour les informations d'accès aux tables.
- En mode parallèle, l'utilitaire asntdiff ne prend pas en charge une combinaison d'ordre croissant et décroissant dans la clause ORDER BY. La combinaison doit être la même que dans l'index. Toutefois, l'utilitaire peut ne pas vous offrir des performances optimales lorsque l'index utilise cette combinaison. Les résultats seront malgré tout corrects.
- Pour avoir des performances optimales :
 - Augmentez la limite des ressources système pour les filières d'application et définissez NUMTHREADS sur 21.
 - N'utilisez pas d'alias ou d'expressions de colonnes au niveau des colonnes clés dans SOURCE_SELECT et TARGET_SELECT.

Utilitaire de réparation des tables (**asntrep**)

La commande **asntrep** répare les différences entre les tables source et cible sur tous les serveurs DB2 en supprimant, insérant et en mettant à jour les lignes. La commande est exécutée sur les systèmes d'exploitation Linux, UNIX ou Windows.

La commande **asntrep** utilise la table des différences qui est générée par la commande **asntdiff** pour effectuer les opérations suivantes :

- Supprimer les lignes de la table cible ne possédant aucune clé identique dans la table source
- Insérer des lignes présentes dans la table source, mais ne possédant aucune clé identique dans la table cible
- Mettre à jour des lignes cible possédant des clés identiques dans la table source, mais des données non clés différentes

Pour la réplication Q, la cible doit être une table et non une procédure stockée. Pour la réplication SQL, la cible doit être une table utilisateur, une table des points de cohérence, une table réplique ou une table de copie utilisateur. Si vous associez l'utilitaire **asntrep** à un abonnement Q pour une réplication entre homologues, vous devez réparer toutes les copies d'une table logique à raison de deux copies à la fois.

Exécutez la commande **asntrep** après la commande **asntdiff**. La commande **asntrep** copie la table des différences de la base de données ou du sous-système source vers la cible, puis utilise la copie pour réparer la table cible.

Pour utiliser la commande **asntrep**, vous spécifiez la même clause WHERE que celle utilisée pour la commande **asntdiff** pour identifier le membre de l'ensemble d'abonnements Q ou d'abonnements qui contient les tables source et cible que vous souhaitez synchroniser. Contrairement à l'utilitaire de comparaison, l'utilitaire de réparation ne prend pas en charge l'utilisation d'un fichier en entrée.

Durant le processus de réparation, les contraintes d'intégrité référentielle imposées à la table cible ne sont pas supprimées. Une tentative d'insertion ou de suppression de ligne d'une table cible peut échouer si l'opération enfreint une contrainte d'intégrité référentielle. Aussi, une ligne source en double peut s'avérer impossible à réparer si la cible contient un index à entrées uniques.

Gestion des fonctions de compatibilité SQL DB2 par l'utilitaire de comparaison

DB2 pour Linux, UNIX et Windows version 9.7 offre des améliorations relatives à la compatibilité SQL, telles que les horodatages de longueur variable, le type de données VARCHAR2 avec traitement de chaînes de caractères spéciaux et la compatibilité DATE-TIMESTAMP. Certains éléments doivent être pris en compte pour pouvoir utiliser la commande **asntdiff** avec ces nouvelles fonctions.

Les sections suivantes décrivent les éléments à prendre en compte :

- «Comparaison de colonnes non clé TIMESTAMP dotées de précisions différentes», à la page 351
- «Comparaison de colonnes clé TIMESTAMP dotées de précisions différentes», à la page 351
- «Éléments à prendre en compte lors de l'utilisation des types de données DATE et TIMESTAMP(0)», à la page 353

- «Comportement lors de l'utilisation du paramètre rangecol », à la page 352
- «Option de compatibilité pour les chaînes de texte », à la page 353
- «Option de fichier asntdiff (asntdiff -f) », à la page 354

Comparaison de colonnes non clé **TIMESTAMP** dotées de précisions différentes

Lorsque l'utilitaire asntdiff compare deux tables dont les colonnes **TIMESTAMP** n'ont pas la même précision, il crée une version tronquée de la colonne **TIMESTAMP** la plus longue, puis il compare les deux valeurs de même longueur.

Dans l'exemple suivant, la table A et la table B ont des colonnes **TIMESTAMP** de longueurs différentes :

Table A	Table B
Co11 - TIMESTAMP (6) 2009-02-05-12.46.01.126412	Co11 - TIMESTAMP (12) 2009-02-05-12.46.01.126412000000

Dans ce cas, l'utilitaire asntdiff compare la valeur 2009-02-05-12.46.01.126412 de la table A avec la valeur tronquée de 2009-02-05-12.46.01.126412 de la table B, et signale ensuite les valeurs correspondantes.

Dans l'exemple suivant, la colonne **TIMESTAMP** de la table A est plus longue que celle de la table B car la valeur cible a été tronquée suite à la réplication (cela survient généralement lorsque la base de données cible est d'une version antérieure à 9.7 et prend uniquement en charge la précision **TIMESTAMP** par défaut correspondant à six chiffres) :

Table A	Table B
Co11 - TIMESTAMP (12) 2009-02-05-12.46.01.126412123456	Co11 - TIMESTAMP (6) 2009-02-05-12.46.01.126412

Ici, asntdiff compare une version tronquée de la valeur source, 2009-02-05-12.46.01.126412, avec la valeur cible de 2009-02-05-12.46.01.126412 et signale une correspondance. Chaque fois que asntdiff tronque une colonne **TIMESTAMP**, l'utilitaire émet un message d'avertissement ASN4034W.

Comparaison de colonnes clé **TIMESTAMP** dotées de précisions différentes

Lorsque asntdiff compare des colonnes clé dotées d'une précision **TIMESTAMP** différente, on note les mêmes concepts de base : une version de la colonne **TIMESTAMP** la plus longue est tronquée sur la longueur de la colonne **TIMESTAMP** la plus courte, dans un souci de comparaison.

Dans l'exemple suivant, la table A a une colonne clé **TIMESTAMP**(6) et une colonne de caractères, et la table B a une colonne clé **TIMESTAMP**(12) et une colonne de caractères.

Table A	Table B
KEYCo11 - TIMESTAMP (6) 2009-02-05-12.46.01.126412	KEYCo11 - TIMESTAMP (12) 2009-02-05-12.46.01.126412000000

Table A	Table B
Col2 CHAR(12) "chaîne de test"	Col2 CHAR(12) "test de chaîne"

L'utilitaire compare la valeur clé de la table A, 2009-02-05-12.46.01.126412, avec une version tronquée de la valeur clé de la table B, 2009-02-05-12.46.01.126412, puis il signale une correspondance. Il compare ensuite les valeurs des colonnes non clé "chaîne de test" et "test de chaîne", puis indique un "U 2" (mise à jour requise) dans la table des différences pour indiquer qu'il existe des lignes contenant la même valeur clé dans la source et la cible, mais qu'au moins une colonne non clé est différente dans la cible :

DIFF	Col1 TIMESTAMP(6)
U 2	2009-02-05-12.46.01.126412

La seconde colonne de la table des différences contient toujours la valeur clé. Dans la mesure où la table des différences DDL est basée sur la table source, asntdiff utilise la valeur TIMESTAMP(6) de la source. Si la table source possède la colonne TIMESTAMP la plus longue, par exemple TIMESTAMP(12), l'utilitaire tronque TIMESTAMP(12) sur TIMESTAMP(6) pour comparer les clés. Il utilise toutefois la définition TIMESTAMP(12) de la table source pour créer la table des différences. La valeur clé écrite dans la table des différences est alors la valeur clé utilisée lors de la comparaison : TIMESTAMP(6). Cette valeur est ensuite complétée en TIMESTAMP(12).

Dans ce cas, lorsque vous utilisez l'utilitaire asntrep pour réparer les différences entre les tables source et cible, asntrep suppose que la valeur de la colonne clé cible est le résultat de la réplication, et donc si DB2 complète avec 0s, une clé correspondante du côté cible est découverte et peut être mise à jour.

Comportement lors de l'utilisation du paramètre rangecol

Le paramètre d'appel **rangecol** de l'utilitaire asntdiff, qui permet de comparer un sous-ensemble de lignes dans deux tables basées sur une colonne TIMESTAMP spécifiée, demande également une attention particulière lorsque la colonne TIMESTAMP est de longueur variable et est également une colonne clé.

Table A

KEYCol1 - TIMESTAMP(12)	Col2 CHAR(12)
2009-02-05-12.46.01.126412123456	"chaîne de test"
2009-02-05-12.46.02.126412123456	"chaîne de test"
2009-02-05-12.46.03.126412123456	"chaîne de test"

Table B

KEYCol1 - TIMESTAMP(6)	Col2 CHAR(12)
2009-02-05-12.46.01.126412	"test de chaîne"
2009-02-05-12.46.02.126412	"test de chaîne"
2009-02-05-12.46.03.126412	"test de chaîne"

A l'aide de la table A et de la table B utilisées comme exemples, examinez la portion rangecol suivante issue d'un appel asntdiff dans lequel la précision TIMESTAMP(6) est utilisée pour préciser les lignes à comparer :

```
RANGECOL='KEYCo11' FROM: '2009-02-05-12.46.01.126412'
TO: '2009-02-05-12.46.03.126412'
```

La clause plage est réécrite par l'utilitaire asntdiff dans une instruction SQL SELECT avec une clause BETWEEN :

```
WHERE ("KEYCo11" BETWEEN '2009-02-05-12.46.01.126412' AND
'2009-02-05-12.46.03.126412')
```

Pour inclure toutes les lignes dans le scénario ci-dessus, utilisez les valeurs clé source dans la clause plage. En règle générale, utilisez toujours la valeur de la colonne TIMESTAMP la plus longue dans la clause plage. Par exemple, l'instruction suivante prend en compte les six lignes à la fois du côté source, et du côté cible :

```
RANGECOL='KEYCo11' FROM: '2009-02-05-12.46.01.126412123456'
TO: '2009-02-05-12.46.03.126412123456'
```

Remarque : les scénarios décrits sont applicables uniquement lorsque le contenu de la table cible a été rempli exclusivement par le programme Q Apply ou Apply. Toute interaction manuelle avec la table cible peut aboutir à des résultats imprévus de l'utilitaire asntdiff. Comme toujours, une analyse approfondie des résultats dans la table des différences est obligatoire avant d'utiliser la commande **asntrep** pour réparer les différences.

Éléments à prendre en compte lors de l'utilisation des types de données DATE et TIMESTAMP(0)

L'utilitaire asntdiff ne prend pas en charge la comparaison des types de données DATE et TIMESTAMP(0). Si la fonction de compatibilité du type de données DATE n'est pas activée pour la base de données qui présente la table avec la colonne DATE, l'utilitaire asntdiff émet le message suivant et prend fin de manière anormale : "ASN4003E Le type de données ou la longueur ... sont incompatibles."

L'exemple suivant montre deux bases de données, la seconde étant activée pour utiliser les colonnes TIMESTAMP(0) pour les dates :

Base de données 1, Table A	Base de données 2 (vecteur de compatibilité 0x40), Table B
DATE 02/05/2009	TIMESTAMP(0) 2009-02-05-12.46.01

Pour comparer ces deux tables, vous devez utiliser l'option de fichier asntdiff et appliquer manuellement l'un des deux types de données à un type de données compatible.

Option de compatibilité pour les chaînes de texte

Si l'option de compatibilité des données de type caractères est activée, l'insertion d'une chaîne vide dans une colonne de texte aboutit à une valeur NULL.

Base de données 1, Table A KEYCo11 - TIMESTAMP(6)	Base de données 1, Table A Col2 VARCHAR(12)
2009-02-05-12.46.01.126412	""

Base de données 2 (vecteur de compatibilité 0x20), Table B KEYCo11 - TIMESTAMP(6)	Base de données 2 (vecteur de compatibilité 0x20), Table B Co12 VARCHAR(12)
2009-02-05-12.46.01.126412	NULL

Par défaut, asntdiff indique une différence dans Col2 et signale la nécessité d'une mise à jour dans la table des différences. Si vous ne voulez pas que asntdiff le signale comme étant une différence, vous pouvez utiliser l'option de fichier asntdiff avec l'instruction SQL suivante dans le paramètre SOURCE_SELECT :

```
SELECT Co12 CASE WHEN Co12 = '' THEN NULL ELSE Co12 END FROM Database1
```

Dans ce cas, le message d'avertissement ASN4035W est émis une fois pour vous informer de ce scénario.

Option de fichier asntdiff (asntdiff -f)

Pour remplacer l'un des comportements par défaut mentionnés ci-dessus, il est recommandé d'utiliser l'option de fichier asntdiff présentée dans la version 9.7.

L'option vous permet d'utiliser une expression SQL. Vous pouvez par exemple utiliser une instruction CAST pour éviter la troncature lors de la comparaison de colonnes TIMESTAMP de longueurs différentes.

L'exemple suivant complète TIMESTAMP(6) en TIMESTAMP(12) :

```
SOURCE_SELECT= "SELECT CAST(KEYCo11 AS TIMESTAMP(12) ) AS KEYCo11, Co12  
FROM TABLE_A ORDER BY 1"  
TARGET_SELECT= "SELECT KEYCo11, Co12 FROM TABLE_B ORDER BY 1"
```

Chapitre 22. Utilisation des services système pour exécuter des programmes de réplication Q et de publication d'événements

Vous pouvez exécuter les programmes de réplication pour la réplication Q et la publication d'événements en utilisant des services système conçus pour chaque système d'exploitation.

Le système d'exploitation z/OS peut utiliser le langage JCL, les tâches démarrant avec le système ou le gestionnaire de redémarrage automatique (ARM) pour exécuter les programmes de réplication. Le système d'exploitation Windows peut exécuter des programmes de réplication en utilisant un service système. Vous pouvez planifier des programmes de réplication sur les systèmes d'exploitation Linux, UNIX, Windows et z/OS.

Utilisation des services système z/OS pour exécuter les programmes de réplication Q et de publication d'événements

Sous z/OS, vous pouvez démarrer les programmes de réplication à l'aide du langage JCL ou comme tâches lancées par le système. Vous pouvez utiliser l'ARM (Automatic Restart Manager) pour redémarrer des programmes de réplication ayant échoué.

Exécution des programmes de réplication Q et de publication d'événements à l'aide de JCL

Lors de la réplication de données sous z/OS, vous pouvez utiliser JCL pour faire fonctionner les programmes de réplication.

Spécification du paramètre **CAPTURE_PATH** (z/OS)

Avant de pouvoir exécuter le programme Q Capture à l'aide du langage JCL, vous devez spécifier le paramètre **CAPTURE_PATH** qui contient le chemin d'accès faisant référence au fichier dans lequel le journal de transactions est stocké.

A propos de cette tâche

Si vous ne définissez pas le paramètre, le programme Q Capture écrit des fichiers journaux dans le répertoire principal de l'utilisateur qui soumet le JCL.

Procédure

Pour spécifier le paramètre **CAPTURE_PATH** sous z/OS, utilisez l'une des méthodes suivantes :

JCL Utilisez la zone PARM de l'instruction JCL destinée à démarrer le programme Q Capture. Par exemple :

```
// PARM='/CAPTURE_SERVER=DSN7 CAPTURE_PATH=//JAYQC // LOGSTDOUT  
capture_schema=JAY'
```

Dans cet exemple, le programme Q Capture écrit ses fichiers journaux dans le fichier UTILISATEUR1.JAYQC.D7DP.JAY.QCAP.LOG. USER1 est l'utilisateur qui soumet le JCL.

Si vous voulez attribuer à l'ensemble de fichiers journaux un qualificatif de haut niveau spécifique, suivez cet exemple :

```
// PARM='/capture_server=DSN7 capture_schema=JAY //  
CAPTURE_PATH=//'OEUSR01'
```

Désormais, le programme Q Capture écrit ses fichiers journaux dans le fichier OEUSR01.DSN7.JAY.QCAP.LOG.

Si vous voulez indiquer le chemin d'accès à SYSADM.XYZ, inspirez-vous d'un des exemples suivants :

```
// PARM='/CAPTURE_server=DSN7 Capture_path=//'SYSADM.XYZ //  
capture_schema=JAY'  
// PARM='/CAPTURE_server=DSN7 capture_schema=JAY //  
capture_PATH=//'SYSADM.XYZ'
```

Assurez-vous que le nom de chemin ne dépasse pas la limite de 44 caractères pour les fichiers MVS. L'ID utilisateur qui exécute ce JCL doit être autorisé à écrire dans le fichier précédent.

SQL Emettez une instruction d'insertion dans la table IBMQREP_CAPPARMS. Par exemple :

```
INSERT INTO JAY.IBMQREP_CAPPARMS (qmgr, restartq, adminq, startmode,  
memory_limit, commit_interval, autostop, monitor_interval,monitor_limit,  
trace_limit, signal_limit, prune_interval, sleep_interval, logreuse,  
logstdout, term, capture_path, arch_level)  
VALUES  
( 'CSQ1', 'IBMQREP.ASN.RESTARTQ','IBMQREP.ASN.ADMINQ',  
'WARMSI', 32, 500, 'N', 300000, 10080, 10080, 10080, 300, 5000,  
'N', 'N', 'Y', '///JAYQC', '901') ;
```

Si vous voulez attribuer à l'ensemble de fichiers journaux un qualificatif de haut niveau spécifique, suivez cet exemple :

```
INSERT INTO JAY.IBMQREP_CAPPARMS (qmgr, restartq, adminq, startmode,  
memory_limit, commit_interval, autostop, monitor_interval,monitor_limit,  
trace_limit, signal_limit, prune_interval, sleep_interval, logreuse,  
logstdout, term, capture_path, arch_level)  
VALUES  
( 'CSQ1', 'IBMQREP.ASN.RESTARTQ', 'IBMQREP.ASN.ADMINQ',  
'WARMSI', 32, 500, 'N',300000, 10080, 10080, 10080, 300, 5000, 'N',  
'N', 'Y', '///'OEUSR01','901');
```

Pour indiquer le chemin d'accès à SYSADM.XYZ, suivez cet exemple :

```
INSERT INTO JAY.IBMQREP_CAPPARMS (qmgr, restartq, adminq, startmode,  
memory_limit, commit_interval, autostop, monitor_interval,monitor_limit,  
trace_limit, signal_limit, prune_interval, sleep_interval, logreuse,  
logstdout, term, capture_path, arch_level)  
VALUES  
( 'CSQ1', 'IBMQREP.ASN.RESTARTQ','IBMQREP.ASN.ADMINQ',  
'WARMSI', 32, 500, 'N',300, 10080, 10080, 10080, 300, 5000,  
'N', 'N', 'Y', '///'SYSADM.XYZ','901');
```

Démarrage du programme Q Capture avec JCL

La bibliothèque d'exemples de WebSphere Replication Server pour z/OS version 9 contient des exemples de code JCL et de scripts que vous pouvez modifier et utiliser pour démarrer le programme Q Capture.

Recommandation : Copiez les travaux de la bibliothèque SASNSAMP vers une autre bibliothèque avant d'effectuer les modifications. Pour une liste complète des modèles de travaux figurant dans la bibliothèque, reportez-vous au répertoire du programme.

Procédure

Pour démarrer un programme Q Capture à l'aide de JCL :

1. Indiquez les paramètres facultatifs appropriés dans la zone PARM du travail Q Capture.

Vou devez définir les variables d'environnement TZ (fuseau horaire) et LANG (langue) dans le langage JCL si vous ne les avez pas définies dans le fichier /etc/profile du système ou dans le fichier de profil du répertoire de base du programme de réplication en cours d'exécution. Pour plus d'informations sur la définition de ces variables, voir Installation et personnalisation de la réplication pour z/OS. L'exemple suivant du langage JCL d'appel comporte une définition des variables TZ et LANG :

```
//CAPJFA EXEC PGM=ASNQCAP,  
// PARM='ENVAR('TZ=PST8PDT','LANG=en_US')/CAPTURE_SERVER=DQRG  
// capture_schema=JFA'
```

2. Indiquez un chemin de répertoire avec la variable d'environnement TMPDIR si vous souhaitez que les programmes de réplication écrivent des fichiers temporaires dans un autre répertoire que le répertoire /tmp.

Démarrage du programme Q Apply à l'aide de JCL

La bibliothèque d'exemples de WebSphere Replication Server pour z/OS version 9 contient des exemples de code JCL et de scripts que vous pouvez modifier et utiliser pour démarrer le programme Q Apply.

Recommandation : Copiez les travaux de la bibliothèque SASNSAMP vers une autre bibliothèque avant d'effectuer les modifications. Pour une liste complète des modèles de travaux figurant dans la bibliothèque, reportez-vous au répertoire du programme.

Procédure

Pour démarrer un programme Q Apply à l'aide de JCL :

Indiquez les paramètres facultatifs appropriés dans la zone PARM du travail Q Apply. L'exemple suivant présente le code JCL d'appel pour le programme Q Apply :

```
//PLS EXEC PGM=ASNQAPP, // PARM='APPLY_SERVER=DQRG APPLY_SCHEMA=JAY'
```

Démarrage du moniteur d'alertes de réplication avec JCL

La bibliothèque d'exemples contient des exemples de code JCL et de scripts que vous pouvez modifier et utiliser pour démarrer le moniteur d'alertes de réplication.

Recommandation : Copiez les travaux de la bibliothèque SASNSAMP vers une autre bibliothèque avant d'effectuer les modifications. Pour une liste complète des modèles de travaux figurant dans la bibliothèque, reportez-vous au répertoire du programme.

Procédure

Pour démarrer le moniteur d'alertes de réplication à l'aide de JCL :

Indiquez les paramètres d'appel facultatifs appropriés dans la zone PARM du travail Moniteur d'alertes de réplication. L'exemple suivant présente le code JCL d'appel pour le moniteur d'alertes de réplication :

```
//monasn EXEC PGM=ASNMON,PARM='monitor_server=DSN
monitor_qual=monqual'
```

Exécution des programmes de réplication Q et de publication d'événements avec JCL en mode de traitement par lot

Pour exécuter les programmes Q Capture, Q Apply et de moniteur d'alertes de réplication avec JCL en mode de traitement par lots, vous personnalisez le code JCL dans la bibliothèque SASNSAMP pour le programme approprié.

Procédure

Pour exécuter des programmes de réplication en mode de traitement par lots :

1. Personnalisez le code JCL dans la bibliothèque SASNSAMP pour le programme approprié. Le tableau 29 indique le modèle de travail à utiliser pour démarrer chaque programme :

Tableau 29. Modèles de travaux pour démarrer les programmes de réplication en JCL

Modèle	Programme
ASNQSTRA	Q Apply
ASNQSTRC	Q Capture
ASNSTRM	moniteur d'alertes de réplication
ASNQTON	Trace (pour le programme Q Capture ou Q Apply)

2. Préparez le langage JCL pour z/OS en indiquant les paramètres d'appel facultatifs appropriés dans la zone PARM des travaux DPROPR (Q Capture, Q Apply, moniteur d'alertes de réplication et asntrc).

Travailler avec les programmes de réplication Q et de publication d'événements en cours de fonctionnement à l'aide de la commande MVS MODIFY

Après avoir démarré le programme Q Capture, le programme Q Apply ou le moniteur d'alertes de réplication, vous pouvez utiliser la commande **MODIFY** pour arrêter le programme ou effectuer des tâches associées.

A propos de cette tâche

Pour des descriptions des paramètres facultatifs que vous pouvez utiliser avec la commande **MODIFY**, voir `asnqccmd` : Utilisation d'un programme Q Capture en cours d'exécution, `asnqacmd` : Utilisation d'un programme Q Apply en cours d'exécution et `asnmcmd` : utilisation d'un moniteur d'alertes de réplication en cours de fonctionnement.

Procédure

Gérer les programmes en cours d'exécution sur z/OS :

Exécutez la commande **MODIFY** à partir de la console z/OS. Vous pouvez utiliser l'abréviation *f*, comme indiqué dans l'exemple de syntaxe ci-dessous :

►► *f*—*nom de travail*—, — | Paramètres | —————►►

f nomtravail , remplace le nom de la commande réelle : **asnqccmd**, **asnqacmd** ou **asnmcmd**. Les paramètres opérationnels qui s'appliquent à chacune des commandes peuvent être utilisés avec le mot clé *f*.

Par exemple, pour arrêter un programme Q Apply en cours de fonctionnement qui utilise le nom de travail PLS, vous devez utiliser la commande suivante :

F PLS,stop

Le tableau 30 répertorie les commandes Q Capture que vous pouvez exécuter avec le mot clé f. Dans tous les exemples, le nom de travail est myqcap.

Tableau 30. Exemples de commandes MODIFY pour le programme Q Capture

Paramètre	Exemple de commande utilisant le mot clé f
prune	f myqcap,prune
qrypargs	f myqcap,qrypargs
reinit	f myqcap,reinit
reinitq	f myqcap,reinitq= <i>nom_file_envoi</i>
startq	f myqcap,startq= <i>nom_file_envoi</i>
startq all	f myqcap,startq all
stopq	f myqcap,stopq= <i>nom_file_envoi</i>
status	f myqcap,status
status show details	f myqcap,status show details
stop	f myqcap,stop
chgparms	f myqcap,chgparms autostop=y/n f myqcap,chgparms commit_interval=n f myqcap,chgparms logreuse=y/n f myqcap,chgparms logstdout=y/n f myqcap,chgparms monitor_interval=n f myqcap,chgparms monitor_limit=n f myqcap,chgparms prune_interval=n f myqcap,chgparms qfull_num_retries=n f myqcap,chgparms qfull_retry_delay=n f myqcap,chgparms sleep_interval=n f myqcap,chgparms signal_limit=n f myqcap,chgparms term=y/n f myqcap,chgparms trace_limit=n

Le tableau 31 répertorie les commandes Q Apply que vous pouvez exécuter avec le mot clé f. Dans tous les exemples, le nom de travail est myqapp.

Tableau 31. Exemples de commandes MODIFY pour le programme Q Apply

Paramètre	Exemple de commande utilisant le mot clé f
loaddonesub	f myqapp,loaddonesub= <i>nom_file_réception:nom_abonnement_q</i>
prune	f myqapp,prune
qrypargs	f myqapp,qrypargs
stopq	f myqapp,stopq= <i>nom_file_réception</i>
startq	f myqapp,startq= <i>nom_file_réception</i>
startq;skiptrans	f myqapp,startq=" <i>nom_file_réception;skiptrans=ID_transaction</i> "
reinitq	f myqapp,reinit= <i>nom_file_réception</i>
stop	f myqapp,stop
status	f myqapp,status
status show details	f myqapp,status show details

Tableau 31. Exemples de commandes MODIFY pour le programme Q Apply (suite)

Paramètre	Exemple de commande utilisant le mot clé f
spillsub	f myqapp,spillsub=nom_file_réception:nom_abonnement_q
resumesub	f myqapp,resumesub=nom_file_réception:nom_abonnement_q
chgparms	f myqapp,chgparms autostop=y/n f myqapp,chgparms logreuse=y/n f myqapp,chgparms logstdout=y/n f myqapp,chgparms monitor_interval=n f myqapp,chgparms monitor_limit=n f myqapp,chgparms prune_interval=n f myqapp,chgparms term=y/n f myqapp,chgparms trace_limit=n f myqapp,chgparms deadlock_retries=n f myqapp,chgparms discardconflicts=y/n f myqapp,chgparms dropris=y/n

Le tableau 32 répertorie les commandes du programme asntrc que vous pouvez exécuter avec le mot clé f. Dans tous les exemples, le nom de travail est myqcap.

Tableau 32. Exemple de commandes MODIFY pour le programme asntrc

Tâche	Exemple de commande utilisant le mot clé f
Démarrez une trace de programme avec la commande asntrc	f myqcap,asntrc on f myqcap,asntrc statlong
Formatez un rapport asntrc fmt et envoyez la sortie à un ensemble de données z/OS	F myqcap, asntrc fmt -ofn // 'USRT001.TRCFMT'
Formatez un rapport asntrc flw et envoyez la sortie à un ensemble de données z/OS	F myqcap, asntrc flw -ofn // 'USRT001.TRCFLW'
Arrêtez une trace de programme	F myqcap, asntrc off

Recommandation : Préaffectez les fichiers de sortie asntrc flw et fmt de manière à ce qu'ils soient suffisamment grands pour contenir les rapports asntrc. Utilisez les attributs suivants :

- **Nom d'ensemble de données :** USRT001.TRCFMT ou USRT001.TRCFLW
- **Cylindres principaux alloués :** 2
- **Extensions allouées normales :** 1
- **Classe de données :** Aucune (utilisation courante)
- **Cylindres utilisés :** 2
- **Format d'enregistrement :** Extensions VB utilisées : 1
- **Longueur d'enregistrement :** 1028
- **Taille de bloc :** 6144
- **Premiers cylindres d'extension :** 2
- **Cylindres secondaires :** 1
- **SMS compressibles :** NON

Le tableau 33, à la page 361 répertorie les commandes du moniteur d'alertes de réplication que vous pouvez exécuter avec le mot clé f. Dans tous les exemples, le nom de travail est mymon.

Tableau 33. Exemples de commandes **MODIFY** pour le programme du moniteur d'alertes de réplication

Paramètre	Exemple de commande utilisant le mot clé f
reinit	f mymon,reinit
status	f mymon,status
qryparms	f mymon,qryparms
suspend	f mymon,suspend
resume	f mymon,resume
stop	f mymon,stop
chgparms	f mymon,chgparms monitor_interval= <i>n</i> f mymon,chgparms autoprune=y f mymon,chgparms trace_limit= <i>n</i> f mymon,chgparms alert_prune_limit= <i>n</i> f mymon,chgparms max_notifications_per_alert= <i>n</i> f mymon,chgparms max_notifications_minutes= <i>n</i>

Pour plus d'informations sur la commande **MODIFY**, voir les commandes système *z/OS MVS*.

Exécution des programmes de réplication Q et de publication d'événements avec les tâches démarrées par le système

Vous pouvez utiliser les tâches démarrées par le système pour faire fonctionner le programme Q Capture, le programme Q Apply et le moniteur d'alertes de réplication.

Procédure

Pour démarrer un programme en tant que tâche démarrée par le système pour le système d'exploitation *z/OS* :

1. Créez une procédure (*nomproc*) dans votre PROCLIB.
2. Créez une entrée dans la classe RACF STARTED pour *nomproc*. Cette entrée associe *nomproc* à l'ID utilisateur RACF à utiliser pour démarrer le programme Q Capture. Assurez-vous que l'autorisation DB2 nécessaire est accordée à cet ID utilisateur avant de démarrer le programme.
3. A partir de la console *z/OS*, exécutez la commande **start** *nomproc*. L'exemple de procédure suivant concerne le programme Q Capture :

```
// PARM='CAPTURE_SERVER=DSN7 capture_schema=ASN startmode=cold'  
//STEPLIB DD DSN=qrh1qua1.SASNLOAD,DISP=SHR  
// DD DSN=dsnh1qua1.SDSNLOAD,DISP=SHR  
//* DD DSN=mqh1qua1.SCSQANLE,DISP=SHR  
//* DD DSN=mqh1qua1.SCSQLOAD,DISP=SHR  
//* DD DSN=xm1h1qua1.SIXMMOD1,DISP=SHR  
//CAPSPILL DD DSN=&&CAPSPILL,DISP=(NEW,DELETE,DELETE),  
// UNIT=VIO,SPACE=(CYL,(50,70)),  
// DCB=(RECFM=VB,BLKSIZE=6404)  
//MSGS DD PATH='/usr/lpp/db2repl_10_01/msg/En_US/db2asn.cat'  
//CEEDUMP DD SYSOUT=* //SYSPRINT DD SYSOUT=* //SYSUDUMP DD DUMMY
```

qrh1qua1

Qualificatif de haut niveau de bibliothèque cible de réplication Q

dsnh1qua1

Qualificatif de haut niveau de bibliothèque cible DB2

mqhlqual

Qualificatif de haut niveau de bibliothèque cible WebSphere MQ

xmlhlqual

Qualificatif de haut niveau de bibliothèque cible XML Toolkit

Le JCL qui exécute le programme Q Capture doit ajouter les bibliothèques WebSphere MQ à STEPLIB si elles ne sont pas installées dans LNKLST. Vous pouvez également ajouter les bibliothèques XML Toolkit si vous souhaitez que la réplication utilise ICU au lieu d'UCS pour les conversions de pages de codes.

Utilisation du gestionnaire ARM (Automatic Restart Manager) pour redémarrer automatiquement la réplication et la publication (z/OS)

Vous pouvez utiliser le système de récupération du gestionnaire ARM sur z/OS pour redémarrer les programmes Q Capture, Q Apply, Capture, Apply, et le moniteur d'alertes de réplication.

Avant de commencer

Veillez à ce que le gestionnaire ARM soit installé et à ce que les programmes de réplication soient correctement paramétrés. Pour utiliser le gestionnaire ARM avec un programme de réplication, veillez à ce que le programme soit autorisé par APF. Par exemple, pour utiliser le gestionnaire ARM avec le programme Q Apply, Apply, ou le moniteur d'alertes de réplication, vous devez copier le module de chargement approprié dans la bibliothèque autorisée par APF. (Les programmes Q Capture et Capture doivent être autorisés par APF, que vous utilisiez ou non le gestionnaire ARM.)

A propos de cette tâche

Le gestionnaire ARM est une fonction de reprise z/OS qui permet d'améliorer la disponibilité de travaux par lots spécifiques ou de tâches démarrées. Lorsqu'un travail ou une tâche échoue, ou que le système sur lequel il s'exécute échoue, le gestionnaire ARM peut redémarrer le travail ou la tâche sans intervention de l'opérateur.

Le gestionnaire ARM utilise des noms d'élément pour identifier les applications dans lesquelles il travaille. Chaque application activée par le gestionnaire ARM génère un nom d'élément unique pour elle-même qu'elle utilise dans toute communication avec le gestionnaire ARM. Le gestionnaire ARM suit le nom d'élément et voit sa politique de redémarrage définie en termes de noms d'éléments. Pour plus de détails sur le paramétrage du gestionnaire ARM, voir *z/OS MVS Sysplex Services Guide*.

Procédure

Pour utiliser le gestionnaire ARM afin de redémarrer automatiquement les programmes de réplication et de publication :

1. Lorsque vous configurez le gestionnaire ARM, indiquez l'un des noms d'élément suivants :

Programme	Nom d'élément
Q Capture	ASNQCxxxxxyyy

Programme	Nom d'élément
Q Apply	ASNQAxxxxyyyy
Capture	ASNTC xxxxyyyy
Apply	ASNTA xxxxyyyy
moniteur d'alertes de réplication	ASNAM xxxxyyyy

Où *xxxx* est le nom du sous-système DB2 et *yyyy* est le nom du membre de partage de données (celui-ci est uniquement nécessaire dans les configurations de partage des données). Le nom d'élément mesure toujours 16 caractères de long, rempli avec des blancs.

2. Facultatif : Si vous avez plusieurs instances d'un programme de réplication ou de publication en cours d'exécution dans un membre de partage des données, indiquez le paramètre **arm** lorsque vous démarrez les programmes afin de créer un nom d'élément ARM unique pour chaque instance de programme.

Le paramètre **arm** prend une valeur de trois caractères qui est accolée aux noms d'éléments indiqués dans la table précédente. La syntaxe est **arm=zzz**, où *zzz* peut correspondre à toute longueur d'une chaîne alphanumérique. Le programme de réplication ne concatènera que trois caractères maximum au nom en cours et remplira avec des blancs si nécessaire, pour composer un nom unique de 16 caractères.

Les programmes de réplication utilisent le nom d'élément pour s'enregistrer avec le gestionnaire ARM pendant l'initialisation. Ils ne fournissent pas d'exit d'événement au gestionnaire ARM lorsqu'ils s'enregistrent. L'exit d'événement n'est pas nécessaire car le programme de réplication ne s'exécute pas en tant que sous-système z/OS. Le gestionnaire ARM redémarre les programmes enregistrés s'ils se sont terminés avec une anomalie (par exemple si une violation de segment se produit). Un programme de réplication enregistré se désenregistre s'il s'est terminé normalement (par exemple en raison d'une commande STOP) ou s'il rencontre un enregistrement non valide.

Conseil : Si vous démarrez le programme Q Capture, Q Apply, Capture, Apply, ou le moniteur d'alertes de réplication à l'aide du paramètre **term=n**, le programme ne s'arrête pas lorsque DB2 est mis en veille ou arrêté. Dans ce cas, le programme ne se désenregistre pas du gestionnaire ARM. Il continue de s'exécuter mais n'effectue pas son travail réel tant que DB2 n'est relancé ou redémarré.

Services de réplication (Windows)

Vous pouvez exécuter les programmes de réplication en tant que service système sur le système d'exploitation Windows en utilisant Windows Service Control Manager (SCM).

Description des services Windows pour la réplication

Sur le système d'exploitation Windows, un service de réplication est un programme chargé de lancer et d'arrêter les programmes Q Capture, Q Apply, Capture, Apply ou Moniteur d'alertes de réplication.

Lorsque vous créez un service de réplication, il est ajouté au Gestionnaire de contrôle de service en mode Automatique, puis démarré. Windows enregistre le service sous un nom de service et un nom d'affichage uniques.

Les termes suivants décrivent les règles d'attribution de noms aux services de réplication :

Nom du service de réplication

Le nom du service de réplication identifie de manière unique chaque service, et permet d'arrêter et de démarrer un service. Son format se présente comme suit :

`DB2.instance.alias.programme.qualificatif_ou_schéma`

Le tableau 34 décrit les entrées pour le nom du service de réplication.

Tableau 34. Entrées pour le nom du service de réplication

Entrée	Description
<i>instance</i>	Nom de l'instance DB2.
<i>alias</i>	Alias de la base de données du serveur Q Capture, Q Apply, du serveur de contrôle Capture, Apply ou du serveur de contrôle Moniteur d'alertes de réplication.
<i>programme</i>	L'une des valeurs suivantes : QCAP (pour le programme Q Capture), QAPP (pour le programme Q Apply), CAP (pour le programme Capture), APP (pour le programme Apply) ou MON (pour le programme Moniteur d'alertes de réplication).
<i>qualificatif_ou_schéma</i>	L'un des qualificatifs suivants : schéma Q Capture, schéma Q Apply, schéma Capture, qualificatif Apply ou qualificatif Moniteur.

Exemple : Le nom de service suivant est attribué à un programme Q Apply qui comprend le schéma ASN et qui fonctionne avec la base de données DB1 sous l'instance appelée INST1 :

`DB2.INST1.DB1.QAPP.ASN`

Nom d'affichage du service de réplication

Le nom d'affichage est une chaîne de texte qui apparaît dans la fenêtre Services. Il représente une forme plus lisible du nom de service. Par exemple :

`DB2 - INST1 DB1 QAPPLY ASN`

Si vous souhaitez ajouter une description du service, utilisez le Gestionnaire de contrôle de service (SCM) après avoir créé un service de réplication. Vous pouvez également utiliser le SCM pour attribuer un nom d'utilisateur et un mot de passe à un service.

Création d'un service de réplication

Vous pouvez créer un service de réplication pour lancer un programme Q Capture, Q Apply, Capture, Apply et le Moniteur d'alertes de réplication sur les systèmes d'exploitation Windows.

Avant de commencer

- Avant de créer un service de réplication, assurez-vous que le service d'instance DB2 est activé. Si le service d'instance DB2 n'est pas activé lors de la création du service de réplication, le service sera créé mais pas démarré automatiquement.
- Après l'installation de DB2, vous devez redémarrer votre serveur Windows avant de démarrer un service de réplication.

A propos de cette tâche

Lors de la création d'un service, vous devez spécifier le nom de compte utilisé pour vous connecter à Windows et le mot de passe correspondant.

Vous pouvez ajouter plusieurs services de réplication à votre système. Vous pouvez ajouter un service pour chacun des schémas sur chaque serveur de contrôle Q Capture, Q Apply ou Capture, et pour chacun des qualificatifs sur chaque serveur de contrôle Apply et Moniteur d'alertes de réplication, respectivement. Par exemple, si vous disposez de cinq bases de données et que chacune d'elles est un serveur de contrôle Q Apply et un serveur de contrôle Moniteur d'alertes de réplication, vous pouvez créer dix services de réplication. Si chaque serveur contient plusieurs schémas ou qualificatifs, vous pouvez en créer davantage.

Procédure

Pour créer un service de réplication :

Exécutez la commande **asnsCRT**.

Lors de la création d'un service, vous devez spécifier le nom de compte utilisé pour vous connecter à Windows et le mot de passe correspondant.

Conseil : Si votre service de réplication est correctement configuré, le nom du service est envoyé à stdout une fois le service correctement démarré. Si le service ne démarre pas, recherchez dans les fichiers journaux le programme que vous avez tenté de lancer. Par défaut, les fichiers journaux sont situés dans le répertoire spécifié par la variable d'environnement DB2PATH. Vous pouvez remplacer cette valeur par défaut en spécifiant le paramètre de chemin d'accès (**chemin_capture**, **chemin_apply**, **chemin_moniteur**) pour le programme qui est lancé en tant que service. De plus, vous pouvez utiliser le Gestionnaire de contrôle de service (SCM) Windows pour visualiser l'état du service.

Lancement d'un service de réplication

Après avoir créé un service de réplication, vous pouvez l'arrêter et le redémarrer.

A propos de cette tâche

Important : Si vous avez démarré un programme de réplication à partir d'un service, un message d'erreur s'affichera si vous essayez de lancer le programme à l'aide du même schéma ou qualificatif.

Procédure

Pour démarrer un service de réplication, utilisez l'une des méthodes suivantes.

- Le Gestionnaire de contrôle de service (SCM) Windows
- La commande **net stop**

Arrêt d'un service de réplication

Après avoir créé un service de réplication, vous pouvez l'arrêter et le redémarrer.

A propos de cette tâche

L'arrêt d'un service de réplication entraîne automatiquement celui du programme qui y est associé. Cependant, si vous arrêtez un programme à l'aide d'une

commande du système de réplication (**asnqacmd**, **asnqccmd**, **asnccmd**, **asnacmd** ou **asnmcmd**), le service qui a démarré le programme reste activé. Vous devez l'arrêter explicitement.

Procédure

Pour arrêter un service de réplication, utilisez l'une des méthodes suivantes.

- Le Gestionnaire de contrôle de service (SCM) Windows
- La commande **net stop**

Affichage d'une liste des services de réplication

Vous pouvez afficher une liste de tous vos services de réplication et leurs propriétés à l'aide de la commande **asnlst**.

Procédure

Pour afficher une liste des services de réplication, utilisez la commande **asnlst**. Vous pouvez éventuellement utiliser le paramètre **details** pour afficher une liste des services de réplication et des descriptions de chaque service.

Suppression d'un service de réplication

Si vous n'avez plus besoin d'un service de réplication, vous pouvez le supprimer afin qu'il soit retiré du Gestionnaire de contrôle de service (SCM) Windows.

A propos de cette tâche

Si vous souhaitez modifier les paramètres de démarrage d'un programme lancé par un service, vous devez supprimer le service et en créer un à l'aide des nouveaux paramètres de démarrage.

Procédure

Pour supprimer un service pour des commandes de réplication, exécutez la commande **asnsdrop**.

Planification des programmes de réplication

Vous pouvez planifier le programme Q Capture, le programme Q Apply ou le moniteur d'alerte de réplication pour qu'ils démarrent à des heures définies.

Planification des programmes de publication d'événements et de réplication (Linux, UNIX)

Pour démarrer un programme de réplication à un moment précis sur un système d'exploitation Linux ou UNIX, utilisez la commande **at**.

A propos de cette tâche

Le tableau 35 affiche les commandes utilisées pour démarrer les programmes de réplication à 15:00 le vendredi.

Tableau 35. Planification des commandes pour les programmes de réplication (Linux, UNIX)

Programme de réplication	Commande Linux ou UNIX
Q Capture	at 3pm Friday asnqcap autoprune=n

Tableau 35. Planification des commandes pour les programmes de réplication (Linux, UNIX) (suite)

Programme de réplication	Commande Linux ou UNIX
Q Apply	at 3pm Friday asnqapply applyqual=myqual
moniteur d'alertes de réplication	at 3pm Friday asnmon monitor_server=db2srv1 monitor_qualifier=mymon

Planification des programmes de réplication (Windows)

Vous pouvez utiliser le gestionnaire de contrôle de service Windows, le gestionnaire de tâches Windows ou la commande **at** pour démarrer les programmes de réplication à un moment précis sur des systèmes d'exploitation Windows. Cette rubrique décrit l'utilisation de la commande **at**.

Procédure

Pour démarrer un programme de réplication à un moment précis sur un système d'exploitation Windows :

1. Démarrez le service de planification Windows.
2. Créez un fichier des mots de passe dans le répertoire du programme de réplication (CAPTURE_PATH, APPLY_PATH, or MONITOR_PATH). Le fichier des mots de passe doit contenir des entrées pour les serveurs pour lesquels le programme de réplication que vous démarrez est en cours de fonctionnement.
3. Emettez la commande **at**. Envoyez la sortie vers un fichier pour vérifier les erreurs.

Le tableau 36 affiche les commandes utilisées pour démarrer les programmes de réplication à 15:00 le vendredi. Notez la présence du caractère "^" lors de la redirection vers un fichier.

Tableau 36. Planification des commandes pour les programmes de réplication (Windows)

Programme de réplication	Commande Windows
Q Capture	c:\>at 15:00 db2cmd asnqcap capture_server=qcapdb capture_schema=schema capture_path=c:\capture ^> c:\capture\asnqcap.out
Q Apply	c:\>at 15:00 db2cmd asnqapp apply_server=qappdb apply_schema=applyqual apply_path=c:\apply ^> c:\apply\asnqapp.out
moniteur d'alertes de réplication	c:\>at 15:00 db2cmd asnmon monitor_server=mondb monitor_qual=monqual monitor_path=c:\ monitor ^> c:\monitor\asnmon.out

Planification des programmes de réplication et de publication d'événements (z/OS)

Vous pouvez utiliser la commande **\$TA JES2** ou la commande **AT NetView** pour démarrer les programmes Q Capture et Q Apply à une heure précise sur z/OS.

Procédure

Pour planifier les programmes de réplication et de publication d'événements sur le système d'exploitation z/OS :

1. Créez une procédure qui appelle le programme pour z/OS dans la PROCLIB.
2. Modifiez le module RACF (fonction de contrôle d'accès aux données) (ou les définitions appropriées pour votre progiciel de sécurité MVS) pour associer la procédure à un ID utilisateur.
3. Effectuez une édition de lien du module dans SYS1.LPALIB.
4. Utilisez la commande **\$TA JES2** ou la commande **AT NetView** pour démarrer le programme Q Capture ou le programme Q Apply à une heure précise. Reportez-vous aux *Commandes JES2 MVS/ESA* pour plus d'informations sur l'utilisation de la commande **\$TA JES2**. Reportez-vous au *Guide des commandes NetView pour MVS* pour plus d'informations sur l'utilisation de la commande **AT NetView**.

Chapitre 23. Conventions et instructions de dénomination pour la réplication Q et la publication d'événements - Présentation

Lorsque vous créez des objets pour la réplication Q et la publication d'événements, vous devez respecter certaines restrictions pour les types de caractère et la longueur de chaque nom d'objet.

Vous devez également respecter les caractères minuscules et majuscules.

Conventions de dénomination pour les objets de réplication Q et de publication d'événements

Le nom de chaque objet de réplication Q et de publication d'événements doit respecter des conventions de dénomination.

Le tableau 37 répertorie les limites pour les noms d'objets dans la réplication Q et la publication d'événements.

Tableau 37. Limites pour les noms d'objets dans la réplication Q et la publication d'événements

Objet	Limites des noms	Limite de longueur
Tables source et cible	<p>DB2 : les noms des tables source et des tables cible DB2 doivent respecter les règles d'affectation de nom pour les noms de tables DB2.</p> <p>Autres que DB2 : les noms des tables cible autres que DB2 doivent respecter les règles d'affectation de nom de table requises par le serveur fédéré DB2 pour configurer les pseudonymes.</p>	<p>z/OS</p> <p>Les noms de schéma courts et les noms de schéma longs sont pris en charge pour les tables sous DB2 pour z/OS. Les noms de tables peuvent comprendre jusqu'à :</p> <ul style="list-style-type: none">• 18 octets pour les sous-systèmes exécutant DB2 for z/OS version 8 en mode compatibilité ou version plus ancienne• 128 octets pour les sous-systèmes exécutant DB2 for z/OS Version 8 en mode new-function
		<p>Linux UNIX Windows</p> <p>30 caractères maximum</p>

Tableau 37. Limites pour les noms d'objets dans la réplication Q et la publication d'événements (suite)

Objet	Limites des noms	Limite de longueur
Colonnes source et cible	<p>DB2 : les noms des colonnes source et cible DB2 doivent respecter les conventions de dénomination de colonne DB2.</p> <p>Autres que DB2 : les noms des colonnes cible autres que DB2 doivent respecter les règles d'affectation de nom de colonne requises par le serveur fédéréDB2 pour configurer les pseudonymes.</p>	<p>Cibles de modification cohérente des données (CCD) : un préfixe à un caractère est ajouté aux colonnes d'image avant. Pour éviter l'utilisation de noms de colonnes d'images avant ambigus, vérifiez que les noms de colonnes sources sont tous à 127 caractères et qu'ils ne sont pas en conflit avec des noms de colonnes existants lorsque ce préfixe est ajouté au nom de colonne.</p>
Propriétaire de table	Les noms de schéma courts et les noms de schéma longs sont pris en charge pour les propriétaires de tables.	Les noms de propriétaires de tables peuvent atteindre 128 octets.
File d'attente d'envoi	Le nom de la file d'attente d'envoi peut comporter tous les caractères autorisés par DB2 et WebSphere MQ pour les types de données VARCHAR. Les noms de file d'attente d'envoi ne doivent pas comporter d'espace.	48 caractères maximum
File d'attente de réception	Le nom de la file d'attente de réception peut comporter tous les caractères autorisés par DB2 et WebSphere MQ pour les types de données VARCHAR. Les noms de file d'attente de réception ne doivent pas comporter d'espace.	48 caractères maximum
File d'attente de redémarrage	<p>Le nom de la file de redémarrage peut comporter tous les caractères autorisés par DB2 et WebSphere MQ pour les types de données VARCHAR.</p> <p>Les noms de file d'attente de redémarrage ne doivent pas comporter d'espace.</p>	48 caractères maximum

Tableau 37. Limites pour les noms d'objets dans la réplication Q et la publication d'événements (suite)

Objet	Limites des noms	Limite de longueur
Abonnement Q	<p>Le nom d'un abonnement Q peut comporter tous les caractères autorisés par DB2 pour les colonnes de type de données VARCHAR. Tous les noms d'abonnements Q doivent être uniques. Le nom de l'abonnement Q étant stocké à la fois sur le serveur source et le serveur cible, assurez-vous que le nom est compatible avec les pages de codes sur les deux serveurs.</p> <p>Les noms d'abonnement Q ne doivent pas comporter d'espaces ou de points-virgules (;).</p>	30 caractères maximum
SUBGROUP	<p>Le nom de l'objet SUBGROUP pour la réplication bidirectionnelle et la réplication entre homologues peut inclure tous les caractères autorisés par DB2 pour les colonnes de type de données VARCHAR.</p> <p>Recommandation : Utilisez un nom de groupe unique pour l'ensemble d'abonnements Q d'une table logique.</p>	30 caractères maximum
Publication	<p>Le nom d'une publication peut comporter tous les caractères autorisés par DB2 pour les colonnes de type de données VARCHAR. Pour chaque programme Q Capture, tous les noms de publication doivent être uniques. Assurez-vous que le nom de la publication est compatible avec la page de codes de l'application d'abonnement.</p> <p>Les noms de publication ne doivent pas comporter d'espaces ou de points-virgules (;).</p>	30 caractères maximum
Schéma Q Capture	<p>Le nom du schéma Q Capture peut uniquement comporter les caractères suivants :</p> <ul style="list-style-type: none"> • A à Z (lettres majuscules) • a à z (lettres minuscules) • nombres (0 à 9) • Le caractère de soulignement (_) 	<p>Le schéma Q Capture peut être une chaîne de 128 caractères.</p> <p style="text-align: center;">z/OS</p> <p>Sous-systèmes utilisant la version 8 en mode compatibilité ou version plus ancienne : 18 caractères maximum</p>

Tableau 37. Limites pour les noms d'objets dans la réplication Q et la publication d'événements (suite)

Objet	Limites des noms	Limite de longueur
Schéma Q Apply	<p>Le nom du schéma Q Apply peut uniquement comporter les caractères suivants :</p> <ul style="list-style-type: none"> • A à Z (lettres majuscules) • a à z (lettres minuscules) • nombres (0 à 9) • Le caractère de soulignement (_) 	<p>Le schéma Q Apply peut être une chaîne de 128 caractères.</p> <p>z/OS</p> <p>Sous-systèmes utilisant la version 8 en mode compatibilité ou version plus ancienne : 18 caractères maximum</p>
Qualificatif de moniteur	<p>Le nom du qualificatif de moniteur peut uniquement comporter les caractères suivants :</p> <ul style="list-style-type: none"> • A à Z (lettres majuscules) • a à z (lettres minuscules) • nombres (0 à 9) • Le caractère de soulignement (_) 	<p>Le nom du qualificatif de moniteur peut être une chaîne de 18 caractères maximum.</p>

Gestion des noms d'objet en minuscules pour la réplication Q et la publication

Les commandes système de réplication Q et de publication d'événements, ainsi que le Centre de réplication, par défaut, convertissent en majuscules tous les noms que vous fournissez. Si un nom comporte à la fois des caractères majuscules et minuscules, mettez-le entre guillemets (ou tout autre caractère pour lequel le système cible est configuré) afin de conserver la casse et d'enregistrer le nom exactement comme vous l'avez saisi.

Par exemple, si vous saisissez `myqual` ou `MyQual` ou `MYQUAL`, le nom est enregistré comme `MYQUAL`. Si vous saisissez les mêmes noms et que vous les mettez entre guillemets, ils sont respectivement enregistrés comme `myqual` ou `MyQual` or `MYQUAL`. Certains systèmes d'exploitation ne reconnaissent pas les guillemets et vous pouvez avoir besoin d'utiliser un caractère d'échappement, généralement une barre oblique inversée (`\`).

Windows **Important** : lorsque vous configurez les services Windows pour le programme Q Capture, le programme Q Apply ou le moniteur d'alertes de réplication, vous devez utiliser des noms uniques pour le schéma Q Capture, le schéma Q Apply et le qualificatif de moniteur. Vous ne pouvez pas vous servir de la casse pour différencier les noms. Vous devez utiliser un chemin unique pour distinguer les noms qui sont identiques. Par exemple, supposons que vous avez trois schémas Q Apply : `myschema` , `MySchema` et `MYSHEMA`. Les trois noms utilisent les mêmes caractères mais avec des casses différentes. Si les trois qualificatifs sont dans le même répertoire sur le serveur Q Apply, ils entraîneront des conflits de noms.

Pour les objets WebSphere MQ, toutes les règles d'affectation de nom sont les mêmes que celles définies par WebSphere MQ.

Chapitre 24. Commandes système de réplication Q et de publication d'événements

Pour pouvez exécutez cette commande sous Linux, UNIX, Windows ou UNIX System Services (USS) sous z/OS pour démarrer, faire fonctionner ou modifier les programmes de réplication.

Pour pouvez définir les paramètres dans n'importe quel ordre en tant que paire **nom=valeur**. Les paramètres et leurs arguments ne font pas la distinction entre les majuscules et les minuscules. Utilisez des guillemets doubles pour conserver la casse.

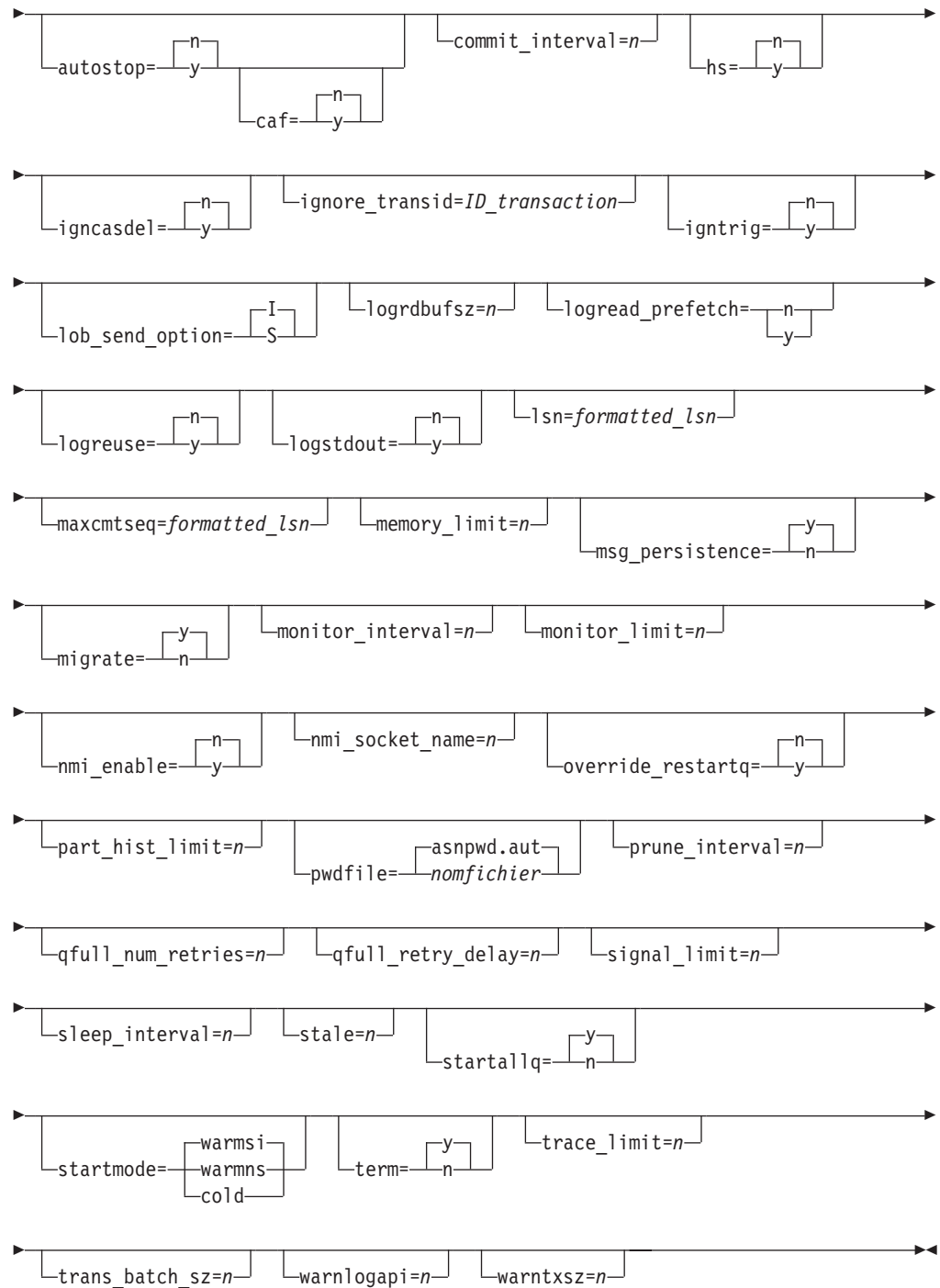
La spécification des paramètres oui/non (booléens) sans argument est tolérée, mais n'est pas recommandée. Par exemple, spécifier **logreuse** donne le même résultat que **logreuse=y**. Mais pour spécifier no logreuse, vous devez utiliser **logreuse=n**.

Appel de commandes avec un point d'interrogation (par exemple, `asnoqcap ?`) affiche un message d'aide montrant la syntaxe de la commande.

Le tableau 38 vous aide à faire la correspondance entre les tâches courantes et les commandes système.

Tableau 38. Tâches de réplication Q et de publication d'événements et commandes système correspondantes

Si vous souhaitez ...	Utilisez cette commande système
Démarrer un programme Q Capture et définir des paramètres de démarrage (Linux, UNIX, Windows, z/OS)	«asnqcap : démarrage d'un programme Q Capture», à la page 374
Démarrer un programme Q Capture pour les sources Oracle et spécifier les paramètres de démarrage Linux, UNIX, Windows)	«asnoqcap : Démarrage d'un programme Q Capture pour une base de données Oracle», à la page 398
Gérer un programme Q Capture en cours d'exécution (Linux, UNIX, Windows, z/OS) <ul style="list-style-type: none">• Vérifier les valeurs de paramètres• Modifier des paramètres• Elaguer les tables de contrôle• Vérifier l'état de Q Capture• Arrêter Q Capture• Réinitialiser tous les abonnements Q ou toutes les publications• Réinitialiser une file d'attente d'envoi	«asnqcmd : utilisation d'un programme Q Capture en cours d'exécution», à la page 407



Descriptions des paramètres asncap

Ces descriptions fournissent des détails sur les paramètres asncap, leurs valeurs par défaut et les raisons susceptibles de vous obliger à modifier la valeur par défaut de votre environnement.

- «Paramètre add_partition (Linux, UNIX, Windows)»
- «arm (z/OS)»
- «autostop», à la page 377
- «caf (z/OS)», à la page 377
- «capture_path», à la page 377
- «capture_schema», à la page 378
- «capture_server», à la page 379
- «commit_interval», à la page 379
- «hs (z/OS)», à la page 380
- «ignore_transid», à la page 380
- «igncasdel», à la page 381
- «ignsetnull», à la page 381
- «igntrig», à la page 382
- «lob_send_option», à la page 382
- «logrdbufsz», à la page 383
- «logread_prefetch (Linux, UNIX et Windows)», à la page 384
- «logreuse», à la page 384
- «logstdout», à la page 385
- «lsn», à la page 385
- «maxcmtseq», à la page 386
- «memory_limit», à la page 387
- «migrate (Linux, UNIX, Windows)», à la page 387
- «monitor_interval», à la page 388
- «msg_persistence», à la page 389
- «override_restartq», à la page 389
- «part_hist_limit (Linux, UNIX, Windows)», à la page 390
- «prune_interval», à la page 390
- «qfull_num_retries», à la page 391
- «qfull_retry_delay», à la page 392
- «signal_limit», à la page 392
- «sleep_interval», à la page 393
- «stale», à la page 393
- «startallq», à la page 394
- «startmode», à la page 394
- «term», à la page 395
- «trace_limit», à la page 396
- «trans_batch_sz», à la page 396
- «warnlogapi (z/OS)», à la page 396
- «warntxsz», à la page 396

Paramètre add_partition (Linux, UNIX, Windows)

Valeur par défaut : add_partition=n

Méthode de modification : au démarrage de Q Capture

Le paramètre **add_partition** spécifie si un programme Q Capture doit commencer à lire le journal de reprise de DB2 pour détecter des partitions ajoutées depuis le dernier redémarrage de Q Capture.

Pour une lecture de ce journal, spécifiez **add_partition=y** au démarrage du programme Q Capture. Lorsque le programme Q Capture est lancé en mode de démarrage à chaud, il lit, pour chaque nouvelle partition, le fichier journal, en commençant par le premier numéro d'ordre (LSN) utilisé par DB2 après l'émission de l'instruction de base de données CONNECT initiale pour l'instance DB2.

Sources Oracle : Le paramètre **add_partition** ne s'applique pas à Q Capture sur les sources Oracle, et n'a aucun effet s'il est spécifié.

arm (z/OS)

Valeur par défaut : aucune

Méthode de modification : au démarrage de Q Capture

Spécifie une chaîne alphanumérique à trois caractères qui est utilisée pour identifier une instance unique du programme Q Capture sur le gestionnaire de redémarrage automatique. La valeur que vous renseignez est associée au nom d'élément ARM que Q Capture génère pour lui-même : ASNQCxxxxxyyyy (où xxxx est le nom de liaison du groupe de partage de données, et yyyy est le nom du

membre DB2). Vous pouvez spécifier toute longueur de chaîne pour le paramètre **arm**, mais le programme Q Capture concaténera uniquement trois caractères maximum au nom en cours. Si nécessaire, le programme Q Capture remplira le nom par des blancs pour composer un nom unique de 16 octets.

autostop

Valeur par défaut : **autostop=n**

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **autostop** détermine si un programme Q Capture s'arrête lorsqu'il atteint la fin du journal DB2 ou redo Oracle actif. Par défaut, un programme Q Capture ne se termine pas en parvenant à la fin de ce fichier.

En règle générale, le programme Q Capture s'exécute en tant que processus continu, que la base de données source soit active ou non, et donc, dans la plupart des cas, il convient de conserver la valeur par défaut (**autostop=n**). Définissez **autostop=y** uniquement dans les cas où le programme Q Capture est exécuté à intervalles réguliers, par exemple, lorsque vous synchronisez des systèmes connectés de manière irrégulière ou dans des scénarios de test.

Si vous définissez **autostop=y**, le programme Q Capture extrait toutes les transactions admissibles et s'arrête lorsqu'il parvient à la fin du journal. Pour extraire d'autres transactions, il vous faut alors redémarrer le programme Q Capture.

caf (z/OS)

Valeur par défaut : **caf =n**

Méthode de modification : au démarrage de Q Capture

Le programme Q Capture s'exécute avec la connexion RRS par défaut (**caf =n**). Vous pouvez remplacer cette valeur par défaut et indiquer au programme Q Capture d'utiliser la fonction de connexion d'appel CAF en définissant l'option **caf =o**. Cette option indique que le programme Q Capture remplace la connexion RRS par défaut et s'exécute avec la connexion CAF.

Si RRS n'est pas disponible, un message vous l'indique et le programme Q utilise la connexion CAF. Le message vous avertit que le programme n'est pas parvenu à se connecter car RRS n'est pas démarré. Le programme tente d'utiliser une connexion CAF à la place. Le programme s'exécute correctement avec la connexion CAF.

capture_path

Valeur par défaut : aucune

Méthodes de modification : au démarrage de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **capture_path** spécifie le répertoire où un programme Q doit stocker ses fichiers de travail et son journal. Par défaut, le chemin est le répertoire dans lequel vous démarrez le programme. Vous pouvez modifier ce chemin.

Vu que le programme Q Capture est une application POSIX, son chemin d'accès par défaut dépend de la manière dont vous démarrez le programme :

- Si vous démarrez un programme Q Capture depuis une invite de ligne de commande USS, le chemin d'accès correspond au répertoire d'où vous lancez le programme.
- Si vous démarrez un programme Q Capture à l'aide d'une tâche démarrée ou via JCL, le chemin par défaut est le répertoire de base dans le système de fichiers USS de l'ID utilisateur associé à la tâche ou au travail démarré.

Pour modifier le chemin, vous pouvez spécifier un autre nom de chemin ou un qualificatif de haut niveau HLQ (high-level qualifier Qualifier), comme //QCAPV9. Lorsque vous utilisez un HLQ, des fichiers séquentiels sont créés conformément aux conventions de dénomination z/OS pour ce type de fichiers. Les fichiers séquentiels dépendent de l'ID utilisateur qui exécute le programme. Sinon, ces noms de fichiers sont identiques à ceux qui sont stockés dans un chemin de répertoire explicitement nommé (le qualificatif HLQ est dans ce cas concaténé en première partie du nom de fichier). Par exemple, sysadm.QCAPV9.nomfichier. L'utilisation d'un qualificatif HLQ peut être utile si vous voulez que les fichiers LOADMSG et les fichiers journaux Q Capture soient gérés par le système (SMS).

Si vous voulez que la tâche lancée par Q Capture écrive sur un ensemble de données .log avec un ID utilisateur différent de l'ID qui exécute la tâche (par exemple TSOUSER), vous devez indiquer une marque de guillemet simple (') comme caractère d'échappement lorsque vous utilisez le format SYSIN pour les paramètres d'entrée sur la tâche démarrée. Par exemple, si vous vouliez utiliser le qualificatif de haut niveau JOESMITH, alors l'ID utilisateur TSOUSER qui exécute le programme Q Capture doit avoir l'autorité RACF pour écrire des fichiers à l'aide du qualificatif de haut niveau JOESMITH, comme dans l'exemple suivant :

```
//SYSIN DD *
  CAPTURE_PATH=/'JOESMITH
/*
```

Windows

Si vous démarrez un programme Q Capture en tant que service Windows, ce programme démarre par défaut depuis le sous-répertoire sql1lib\bin, dans le répertoire d'installation.

capture_schema

Valeur par défaut : `capture_schema=ASN`

Le paramètre **capture_schema** vous permet de faire une distinction entre les diverses instances du programme Q Capture sur un serveur Q Capture.

Le schéma identifie un programme Q Capture spécifique et ses tables de contrôle. Deux programmes Q Capture avec le même schéma ne peuvent pas s'exécuter sur un même serveur.

La création d'une copie supplémentaire d'un programme Q Capture sur un serveur Q Capture vous permet d'améliorer le débit des données en les répartissant en flux parallèles, ou de répondre à des besoins de réplication différents tout en utilisant la même source.

capture_server

z/OS

Valeur par défaut : aucune

Linux UNIX Windows

Valeur par défaut: **capture_server**=valeur de la variable d'environnement DB2DBDFT, si elle est définie

Le paramètre **capture_server** identifie la base de données ou le sous-système sur lequel s'exécute le programme Q Capture et où sont stockées ses tables de contrôle. Ces tables de contrôle contiennent des informations sur les sources, les abonnements Q, les files d'attente MQ de WebSphere et les préférences utilisateur. Dans la mesure où le programme Q Capture lit le journal de la base de données source, il doit s'exécuter sur la base de données ou le sous-système source.

Sources Oracle : Si vous n'indiquez pas de serveur Q Capture, ce paramètre aura comme valeur par défaut la valeur de la variable d'environnement ORACLE_SID.

z/OS

Vous devez définir le paramètre **capture_server**. Pour le partage de données, vous pouvez fournir le nom de liaison du groupe au lieu de celui du sous-système afin de pouvoir exécuter le travail de réplication dans toute partition logique LPAR.

commit_interval

Valeur par défaut : **commit_interval**=500 millisecondes (une demi-seconde) pour les sources DB2 ; 1000 millisecondes (1 seconde) pour les sources Oracle

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **commit_interval** spécifie la fréquence de validation (en millisecondes), par le programme Q Capture, des transactions vers WebSphere MQ. Par défaut, un programme Q Capture patiente 500 millisecondes (une demi-seconde) entre validations. A chaque intervalle, le programme Q Capture émet un appel MQCMIT. Ceci stipule au gestionnaire de files d'attente MQ de WebSphere de rendre disponibles au programme Q Apply, ou à d'autres applications utilisateur, les messages placés dans les files d'attente d'émission.

Chacune des transactions regroupées dans un appel MQCMIT est considérée comme une unité de travail ou une transaction WebSphere MQ. En règle générale, une transaction MQ WebSphere comporte elle-même plusieurs transactions de base de données. Si la transaction de base de données est volumineuse, le programme Q Capture n'émet pas d'appel MQCMIT et ce, même si l'intervalle de validation est atteint. La validation par Q Capture n'intervient qu'après le placement de la transaction de base de données volumineuse dans la file d'attente d'émission.

Lorsque le nombre de transactions de base de données validées lues par un programme Q Capture atteint 128, ce dernier émet un appel MQCMIT quelque soit la valeur du paramètre **commit_interval**.

La recherche de l'intervalle de validation optimal implique un compromis entre le temps d'attente (délai entre la validation dans la base de données source et dans la base de données cible) et le temps système sur l'unité centrale consommé par le processus de validation :

Pour raccourcir ce délai, réduisez l'intervalle de validation

Les transactions seront alors transmises à plus grande fréquence. Ceci est particulièrement important lorsque les modifications de la base de données source servent à déclencher des événements. Si le nombre de transactions publiées par intervalle de publication est élevé, vous devrez sans doute réduire le nombre de transactions validées en même temps par le programme Q Capture vers WebSphere MQ. Pour plus d'informations, voir *Determining the number of transactions published per commit interval*.

Pour réduire la surcharge sur l'unité centrale, rallongez l'intervalle de validation

Un intervalle plus long permet d'envoyer le plus grand nombre possible de transactions de base de données pour chaque transaction MQ WebSphere. Il réduit également le nombre d'E/S induit par la consignation des messages. Si vous devez rallonger l'intervalle de validation, il se peut que vous soyez limité par la mémoire allouée au programme Q Capture, la profondeur maximale (nombre de messages) des files d'attente d'émission et par l'attribut (MAXUMSGS) du gestionnaire de files d'attente déterminant le nombre maximal de messages non validés. Si le programme Q Capture doit patienter plus longtemps entre les validations, la publication de certaines transactions peut être reportée et donc entraîner une augmentation du délai d'attente.

hs (z/OS)

Valeur par défaut : hs=n

Méthode de modification : au démarrage de Q Capture

Le paramètre **hs** indique si Q Capture crée un fichier auxiliaire dans l'hiperspace (espace de données hautes performances) s'il excède sa limite de mémoire au cours d'une tentative d'écriture d'une ligne en mémoire. Par défaut (**hs=n**), Q Capture crée le fichier auxiliaire sur disque ou dans les entrées-sorties virtuelles (VIO).

Recommandation : Allouez une quantité de mémoire suffisante au travail Q Capture pour éviter d'avoir besoin de fichiers auxiliaires.

ignore_transid

Valeur par défaut : aucune

Méthode de modification : au démarrage de Q Capture

Le paramètre **ignore_transid= ID_transaction** indique que le programme Q Capture ignore la transaction identifiée par *ID_transaction*. Les transactions ne sont pas répliquées ni publiées. Vous pouvez utiliser ce paramètre si vous souhaitez ignorer une transaction lourde qui n'a pas besoin d'être répliquée (par exemple, un travail par lots important). La valeur pour *ID_transaction* est un identificateur hexadécimal de 10 octets au format suivant :

z/OS

0000:xxxx:xxxx:xxxx:mmmm

Où *xxxx:xxxx:xxxx* correspond à l'ID de transaction et *mmmm* à l'ID membre pour le partage de données. Vous trouverez l'ID membre dans les deux derniers octets de l'en-tête de l'enregistrement de journal, dans la sortie LOGP. L'ID membre est 0000 si le partage de données n'est pas activé.


```
nnnn:0000:xxxx:xxxx:xxxx
```

où `xxxx:xxxx:xxxx` est l'ID transaction et `nnnn` l'identificateur de partition pour les bases de données partitionnées (0000 pour les bases de données non partitionnées).

Conseil : La version raccourcie **transid** est également acceptable pour ce paramètre.

igncasdel

Valeur par défaut : `igncasdel=n`

Méthode de modification : au démarrage de Q Capture ; au niveau de l'abonnement Q en utilisant des outils d'administration de réplication (valeur enregistrée dans la table `IBMQREP_SUBS`)

Le paramètre **igncasdel** indique si le programme Q Capture doit répliquer des opérations de suppression résultant de la suppression de lignes parentes sur des tables avec des relations d'intégrité référentielle (suppressions en cascade). Cette option permet de réduire la quantité de données requise à propager en cas de suppression en cascade de la ligne parente sur la cible.

Par défaut, lorsqu'une ligne parente est supprimée, Q Capture réplique les opérations de suppression à partir des lignes enfants. Le programme Q Apply ordonne à nouveau aux transactions de vérifier qu'aucune ligne enfant n'est supprimée sur la cible avant que la ligne parente soit elle-même supprimée. Si vous indiquez **igncasdel=y**, Q Capture réplique uniquement la suppression de la ligne parente. Utilisez cette option pour éviter les opérations de suppression redondantes par le programme Q Apply si la réplication de la suppression de la ligne parente est de nature à occasionner des suppressions en cascade dans la table cible.

Vous pouvez également définir cette option au niveau de l'abonnement Q en modifiant la valeur de la colonne `IGNCASDEL` dans la table `IBMQREP_SUBS` à l'aide des outils d'administration de réplication, en la faisant passer de N (par défaut) à Y. Si vous indiquez Y dans la table `IBMQREP_SUBS`, le réglage du paramètre **igncasdel** est remplacé pour l'abonnement Q pris séparément.

ignsetnull

Valeur par défaut : `igntrig=n`

Méthode de modification : au démarrage de Q Capture ; au niveau de l'abonnement Q en utilisant des outils d'administration de réplication (valeur enregistrée dans la table `IBMQREP_SUBS`)

Le paramètre **ignsetnull** indique que le programme Q Capture ne doit pas répliquer des opérations UPDATE résultant de la suppression de lignes parentes dans des tables avec des relations d'intégrité référentielle lorsque la règle ON DELETE SET NULL est appliquée.

Par défaut, lorsqu'une ligne parente est supprimée et que la règle ON DELETE SET NULL est appliquée, Q Capture réplique ces opérations UPDATE dans lesquelles une ou plusieurs valeurs de colonne sont définies sur NULL. Si la règle ON

DELETE SET NULL est appliquée sur la cible, vous pouvez définir **ignsetnull=y** pour que Q Capture ignore ces opérations UPDATE.

Vous pouvez également définir cette option au niveau de l'abonnement Q en modifiant la valeur de la colonne IGNSETNULL dans la table IBMQREP_SUBS à l'aide des outils d'administration de réplication, en la faisant passer de N (par défaut) à Y. Si vous indiquez Y dans la table IBMQREP_SUBS, le réglage du paramètre **ignsetnull** est remplacé pour l'abonnement Q pris séparément.

igntrig

Valeur par défaut : igntrig=n

Méthode de modification : au démarrage de Q Capture ; au niveau de l'abonnement Q en utilisant des outils d'administration de réplication (valeur enregistrée dans la table IBMQREP_SUBS)

Le paramètre **igntrig** indique que le programme Q Capture doit supprimer des lignes générées par des déclencheurs. Si un déclencheur d'une table source génère une instruction SQL secondaire après une opération SQL, les instructions SQL initiale et secondaire sont toutes les deux répliquées dans la cible. Ces instructions secondaires créent des conflits car le déclencheur de la table cible génère les mêmes lignes lorsque des changements de source sont appliqués. Définir **igntrig=y** indique au programme Q Capture de ne pas capturer les instructions SQL générées par des déclencheurs.

Vous pouvez également définir l'option au niveau de l'abonnement Q en modifiant la valeur de la colonne IGNTRIG de la table IBMQREP_SUBS en la faisant passer de n (par défaut) à y. Si vous spécifiez Y dans la table IBMQREP_SUBS, le réglage du paramètre **igntrig** est remplacé pour l'abonnement Q pris séparément.

Si vous utilisez **igntrig=n**, veillez à définir les déclencheurs des tables cible comme identiques à ceux de la table source. Dans le cas contraire, des conflits peuvent se produire lorsque Q Apply met à jour une table cible comportant un déclencheur sans correspondance.

Vous pouvez également définir cette option au niveau de l'abonnement Q en modifiant la valeur de la colonne IGNTRIG dans la table IBMQREP_SUBS à l'aide des outils d'administration de réplication, en la faisant passer de N (par défaut) à Y. Si vous indiquez Y dans la table IBMQREP_SUBS, le réglage du paramètre **igntrig** est remplacé pour l'abonnement Q pris séparément.

Remarque à propos des déclencheurs INSTEAD OF : Si la table source est mise à jour par des déclencheurs INSTEAD OF sur une vue et que vous avez défini le paramètre **igntrig** à y, le programme Q Capture ne réplique pas le changement sur la table source.

lob_send_option

Valeur par défaut : lob_send_option=I

Méthodes de modification : au démarrage de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **lob_send_option** indique si le programme Q Capture envoie des valeurs LOB intégrées (I) au sein du message de transaction ou dans un message

séparé (S). Par défaut, les valeurs d'objets LOB sont envoyées au sein du message de transaction. Le programme Q Capture gère la quantité de mémoire qu'il occupe lorsqu'il réplique ou publie des types de données LOB. Si vous répliquez ou des données LOB, la valeur de **max_message_size** détermine la fréquence à laquelle le programme Q Capture accède à la table source d'extraction de données LOB dans plusieurs paliers (un palier par message). Une petite taille maximale de message peut entraver les performances du programme Q Capture lors de la réplication ou de la publication de données d'objet LOB.

Linux, UNIX, Windows : Démarrage avec DB2 version 10.1, le programme Q Capture peut extraire des données LOB directement à partir du journal de reprise même si les données excèdent la valeur de l'option **INLINE LENGTH** dans la colonne LOB dans une instruction **CREATE TABLE** ou **ALTER TABLE**. Dans des versions précédentes, Q Capture se connectait à la base de données source pour extraire des données LOB si les données étaient plus grandes que l'option **INLINE LENGTH**. Pour plus d'informations, voir Amélioration des performances lors de la réplication de données LOB.

Si vous définissez **lob_send_option=S** afin que les valeurs LOB soient envoyées dans un message LOB séparé, il est possible que vos valeurs LOB ne tiennent pas dans un seul message LOB. Si la taille de la valeur LOB dépasse la taille maximale des messages pour le programme Q Capture, alors le message LOB est alors divisé en au moins deux messages plus petits. Si vous comptez répliquer ou publier plusieurs valeurs LOB ou BLOB, allouez suffisamment de mémoire et d'espace d'archivage, puis définissez la longueur de la file d'attente en conséquence.

Utilisez les directives suivantes pour configurer le paramètre **max_message_size** :

- Une grande transaction typique doit pouvoir être insérée dans un message. Aussi, définissez la valeur du paramètre **max_message_size** légèrement au-dessus de la taille maximale d'une transaction typique.
- Pour les transactions de taille importante dont la valeur excède la valeur **max_message_size**, vérifiez que vous avez défini **max_message_size** de sorte qu'au moins une ligne de la transaction puisse être insérée dans un message.
- La valeur de **max_message_size** doit être inférieure ou égale au paramètre **MAXMSGL**, qui définit la taille maximale des messages d'une file d'attente.

Lorsque le programme Q Capture utilise le mode LOB séparé, les valeurs LOB de toutes les colonnes LOB faisant partie d'un abonnement Q ou d'une publication sont envoyées pour chaque ligne d'une transaction. Ce comportement engendre davantage de messages WebSphere MQ si une valeur LOB est mise à jour à plusieurs reprises dans une transaction ou si l'option **CHANGED_COLS_ONLY** de l'abonnement Q ou de la publication est définie sur N.

Les valeurs LOB de toutes les colonnes LOB faisant partie d'un abonnement Q ou d'une publication sont envoyées pour les mises à jour des clés sans tenir compte de la définition de **CHANGED_COLS_ONLY**.

logrdbufsz

Valeur par défaut : **logrdbufsz=66** Ko pour z/OS ; 256 Ko pour Linux, UNIX et Windows

Méthodes de modification : au démarrage de Q Capture ; table **IBMQREP_CAPPARMS**

Le paramètre **logrdbufsz** spécifie la taille de la mémoire tampon (en Ko) que le programme Q Capture transmet à DB2 lorsque Q Capture extrait les enregistrements de journaux. DB2 remplit la mémoire tampon avec les enregistrements de journaux disponibles que Q Capture n'a pas extrait. Pour les bases de données partitionnées, Q Capture alloue une mémoire tampon de la taille spécifiée par **logrdbufsz** pour chaque partition.

Les valeurs par défaut sont normalement optimales pour la plupart des situations. Toutefois, il est possible que vous souhaitiez augmenter cette valeur si de nombreuses données ont été modifiées et que vous disposez de suffisamment de mémoire.

logread_prefetch (Linux, UNIX et Windows)

Par défaut : **logread_prefetch=y** pour les bases de données partitionnées ; **n** pour les bases de données non partitionnées

Méthode de modification : au démarrage de Q Capture

Le paramètre **logread_prefetch** indique si le programme Q Capture utilise des unités d'exécution séparées pour préextraire les enregistrements de journal depuis chaque partition d'une base de données partitionnée. Par défaut (**logread_prefetch=y**), Q Capture utilise des unités d'exécution du lecteur du journal séparées pour se connecter à chaque partition. L'utilisation d'unités d'exécution séparées peut améliorer la capacité de traitement de Q Capture mais solliciter davantage l'unité centrale.

Si vous indiquez **logread_prefetch=n**, une seule unité d'exécution du lecteur du journal Q Capture se connecte à toutes les partitions.

logreuse

Valeur par défaut : **logreuse=N**

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Chaque programme Q Capture conserve un fichier journal diagnostic pour le suivi de son historique de travail (comme heure de démarrage et d'arrêt, modifications de paramètres, erreurs, élagage et points d'abandon lors de la lecture du journal de base de données).

Par défaut, le programme Q Capture ajoute des informations au journal existant lors de son redémarrage. Cette valeur par défaut permet de conserver un historique des actions du programme. Si vous ne souhaitez pas conserver cet historique ou si vous désirez économiser de l'espace, définissez le paramètre **logreuse=y**. Le programme Q Capture efface le fichier journal lors de son démarrage, puis écrit dans le fichier vierge.

Le journal est conservé par défaut dans le répertoire de démarrage du programme Q Capture, ou à un emplacement différent que vous aurez spécifié à l'aide du paramètre **capture_path**.

z/OS Le nom du fichier journal est *serveur_capture.schéma_capture.QCAP.log*. Par exemple, *SAMPLE.ASN.QCAP.log*. De plus, si **chemin_capture** est spécifié avec des barres obliques (//) afin d'utiliser un

qualificatif de haut niveau (HLQ), les conventions de dénomination de fichier séquentiel z/OS s'appliquent et **schéma_capture** est tronqué à 8 caractères.

Linux UNIX Windows Le nom du fichier journal est *instancedb2.serveur_capture.schéma_capture.QCAP.log*. Par exemple, *DB2.SAMPLE.ASN.QCAP.log*.

Sources Oracle : Le nom du fichier journal est *serveur_capture.schéma_capture.QCAP.log*. Par exemple, *ORASAMPLE.ASN.QCAP.log*.

logstdout

Valeur par défaut : **logstdout=n**

Par défaut, un programme Q Capture consigne son historique de travail uniquement dans son fichier journal. Vous pouvez modifier le paramètre **logstdout** si vous désirez visualiser cet historique depuis la sortie standard (stdout), en plus du journal.

Les messages d'erreur et certains messages du journal (initialisation, arrêt, activation et désactivation d'abonnement) sont consignés à la fois dans la sortie standard et dans le fichier journal quelle que soit la valeur de ce paramètre.

lsn

Valeur par défaut : aucune

Méthode de modification : au démarrage de Q Capture

Le paramètre **lsn** indique le numéro d'ordre du journal auquel le programme Q Capture démarre lors d'un redémarrage à chaud. Vous pouvez spécifier les paramètres **lsn** et **maxcmtseq** pour démarrer Q Capture à partir d'un emplacement connu dans le journal DB2. Lorsque vous définissez le paramètre **lsn**, vous devez également définir le paramètre **maxcmtseq** dans le même appel de commande et vous ne pouvez pas utiliser ces paramètres si la valeur de **startmode** indique un démarrage à froid.

Cette valeur représente le premier LSN que le programme Q Capture a identifié pour lequel aucun enregistrement de validation ou d'abandon n'a encore été trouvé. Vous pouvez obtenir la valeur de **lsn** à partir du message de redémarrage en utilisant la commande **asnqfmt**. Vous pouvez également utiliser la valeur de la colonne **RESTART_SEQ** dans la table **IBMQREP_CAPMON**. Si vous utilisez cette méthode, choisissez une entrée dans la table de contrôle qui est plus ancienne que l'heure d'arrêt du programme Q Capture afin de vous assurer que tous les messages potentiellement perdus seront à nouveau capturés.

Pour démarrer à partir de la fin du journal sans déclencher une charge (régénération intégrale) de la table cible, indiquez l'une des valeurs suivantes, selon votre version de DB2 :

Version 9.7 et précédentes

lsn=FFFF:FFFF:FFFF:FFFF:FFFF et **maxcmtseq=FFFF:FFFF:FFFF:FFFF:FFFF**.

Version 10.1 ou supérieure avec le paramètre compatibility de 1001 ou supérieur, ou version 9.8

lsn=FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF et
maxcmtseq=FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF.

Vous pouvez également définir **lsn** et **maxcmtseq** sans double points, pour économiser de l'espace.

maxcmtseq

Valeur par défaut : aucune

Méthode de modification : au démarrage de Q Capture

Le paramètre **maxcmtseq** est utilisé pour spécifier la position d'enregistrement du journal de validation de la dernière transaction envoyée avec succès par le programme Q Capture avant son arrêt. Vous pouvez spécifier les paramètres **maxcmtseq** et **lsn** pour démarrer Q Capture à partir d'un emplacement connu dans le journal DB2. Lorsque vous définissez le paramètre **maxcmtseq**, vous devez également définir le paramètre **lsn** dans le même appel de commande et vous ne pouvez pas utiliser ces paramètres si la valeur de **startmode** indique un démarrage à froid.

La valeur de **maxcmtseq** est un marqueur de journal interne qui est différent pour chaque type de système de base de données. Le marqueur est codé comme une chaîne de 10 caractères :

z/OS

Sous z/OS, la valeur est le numéro d'ordre du journal (LSN) de l'enregistrement de journal de validation auquel Q Capture peut ajouter un numéro d'ordre, car sous z/OS avec partage de données, différents enregistrements de journaux peuvent avoir le même LSN.

Linux UNIX Windows

Sous Linux, UNIX et Windows, la valeur est un horodatage avec une précision au niveau de la nanoseconde identifiant une transaction de manière unique. La valeur est codée sous la forme de deux entiers, des secondes et des nanosecondes.

Vous pouvez trouver la valeur de **maxcmtseq** en utilisant l'une des méthodes suivantes :

- A partir du message de redémarrage, en utilisant la commande **asnqfmt**
- A partir du fichier journal de sortie Q Capture, dans les messages ASN7108I et ASN7109
- A partir de la table IBMQREP_APPLYMON (OLDEST_COMMIT_LSN pour les sources z/OS et OLDEST_COMMIT_SEQ pour les sources Linux, UNIX et Windows)

Pour démarrer à partir de la fin du journal sans déclencher une charge (régénération intégrale) de la table cible, indiquez l'une des valeurs suivantes, selon votre version de DB2 :

Version 9.7 et précédentes

lsn=FFFF:FFFF:FFFF:FFFF:FFFF et **maxcmtseq=FFFF:FFFF:FFFF:FFFF:FFFF**.

Version 10.1 ou supérieure avec le paramètre **compatibility de 1001 ou supérieur, ou version 9.8**

lsn=FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF et
maxcmtseq=FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF.

Vous pouvez également définir **lsn** et **maxcmtseq** sans double points, pour économiser de l'espace.

memory_limit

Valeur par défaut : `memory_limit=0` sous z/OS ; 500 Mo sous Linux, UNIX et Windows

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre `memory_limit` spécifie la quantité de mémoire utilisable par un programme Q Capture pour créer des transactions de base de données en mémoire. Par défaut, le paramètre `memory_limit` est défini sur 0 sous z/OS et le programme Q Capture calcule une allocation de mémoire basée sur la taille de la région de Q Capture dans JCL ou la tâche démarrée. Sous Linux, UNIX et Windows un programme Q Capture utilise un maximum de 500 Mo par défaut. Lorsque toute la mémoire allouée par ce paramètre est saturée, le programme Q Capture déverse les transactions en mémoire dans un fichier situé dans le répertoire `capture_path`. Sous z/OS, le programme Q Capture déverse vers VIO ou dans le fichier spécifié dans la carte CAPSPILL DD.

La valeur maximale autorisée pour ce paramètre est de 100 Go.

Vous pouvez ajuster cette limite en fonction de vos besoins :

Pour améliorer les performances d'un programme Q Capture, augmentez la limite affectée à cette mémoire

Si votre objectif est d'obtenir un débit plus élevé, maximisez cette limite dans la mesure du possible.

Pour conserver vos ressources système, diminuez cette limite

Une limite de mémoire plus restrictive réduit la concurrence avec les autres opérations du système. Cependant, une limite trop basse utilisera plus d'espace système pour le fichier auxiliaire et induira plus d'E/S pouvant ralentir votre système.

Vous pouvez exploiter les données de la table IBMQREP_CAPMON afin de déterminer la limite de mémoire la mieux adaptée à vos besoins. Vérifiez, par exemple, la valeur de `CURRENT_MEMORY` afin d'établir quelle quantité de mémoire le programme Q Capture utilise pour reconstruire les transactions du journal. Ou bien, vérifiez la valeur de `TRANS_SPILLED` pour déterminer combien de transactions ont été déversées vers un fichier par le programme Q Capture suite à la saturation de sa limite de mémoire. Vous pouvez vérifier ces valeurs depuis la fenêtre Débit Q Capture dans le Centre de réplication. Pour plus de détails, reportez-vous à l'aide en ligne du Centre de réplication.

migrate (Linux, UNIX, Windows)

Valeur par défaut : `migrate=y`

Méthode de modification : au démarrage de Q Capture

Le paramètre `migrate` indique que le programme Q Capture commence à partir du début du journal après la mise à niveau de DB2 for Linux, UNIX et Windows. Utilisez cette option après avoir mis à niveau DB2 sur une nouvelle édition comme la version 9.5 ou 9.7. Le paramètre `migrate` n'est pas obligatoire après la mise à niveau de DB2 avec un groupe de correctifs.

Utilisez le paramètre **migrate** uniquement lors du premier démarrage de Q Capture et indiquez **startmode=warmns**.

monitor_interval

Valeurs par défaut : **monitor_interval**=60000 millisecondes (1 minute) sous z/OS et 30000 millisecondes (30 secondes) sous Linux, UNIX, et Windows

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **monitor_interval** indique au programme Q Capture à quelle fréquence insérer des statistiques de performances dans deux de ses tables de contrôle. La table IBMQREP_CAPMON présente des statistiques de performance globale du programme Q Capture et la table IBMQREP_CAPQMON, des statistiques sur chaque file d'attente d'émission.

Par défaut, des lignes sont insérées dans ces tables toutes les 300 000 millisecondes (5 minutes). Généralement, un programme Q Capture valide les transactions MQ WebSphere à une fréquence beaucoup plus courte (l'intervalle de validation par défaut étant d'une demi-seconde). Par conséquent, si vous utilisez les valeurs par défaut pour l'intervalle de contrôle et de validation, chaque insertion dans les tables représente les totaux relatifs à 120 validations. Si vous souhaitez contrôler le programme Q Capture à un niveau plus granulaire, utilisez un intervalle de contrôle plus proche de l'intervalle de validation.

Important pour les utilisateurs du tableau de bord de réplication Q : Lorsque cela est possible, vous devez synchroniser le paramètre **monitor_interval** de Q Capture avec l'intervalle d'actualisation du tableau de bord (à quelle fréquence le tableau de bord récupère des informations de performance des tables de contrôle Q Capture et Q Apply). L'intervalle d'actualisation par défaut du tableau de bord est de 10 secondes (10000 millisecondes). Si la valeur de **monitor_interval** est supérieure à l'intervalle d'actualisation du tableau de bord, ce dernier s'actualise alors qu'aucune nouvelle donnée de surveillance n'est disponible.

monitor_limit

Valeur par défaut : **monitor_limit**=10 080 minutes (7 jours)

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **monitor_limit** spécifie le degré d'ancienneté requis des lignes des tables IBMQREP_CAPMON et IBMQREP_CAPQMON avant d'être candidates à un élagage.

Par défaut, les lignes datant de plus de 10 080 minutes (7 jours) sont élaguées. Les tables IBMQREP_CAPMON et IBMQREP_CAPQMON contiennent des statistiques sur l'activité d'un programme Q Capture. Une ligne est insérée à chaque intervalle de contrôle. Vous pouvez adapter le nombre maximal de contrôles en fonction de vos besoins :

Augmentez le nombre maximal de contrôles pour conserver plus de statistiques

Si vous désirez conserver une trace de l'activité du programme Q Capture au delà d'une semaine, définissez un nombre maximal de contrôles plus élevé.

Diminuez cette limite si vous consultez vos statistiques fréquemment

Si vous suivez l'activité du programme Q Capture sur une base régulière, vous n'avez probablement pas à conserver une semaine de statistiques et vous pouvez définir une limite plus basse qui induira un élagage plus fréquent.

msg_persistence

Valeur par défaut : `msg_persistence=y`

Méthodes de modification : au démarrage de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **msg_persistence** Indique si un programme Q Capture enregistre des messages permanents (consignés) ou des messages de données non permanents (non consignés) dans des files d'attente WebSphere MQ. (Les messages de données contiennent des données répliquées provenant des tables source.) Par défaut, Q Capture utilise des messages de données permanents. Le gestionnaire de files d'attente consigne les messages permanents et peut récupérer les messages après un redémarrage ou une défaillance système.

Dans certains cas, il est possible que vous souhaitiez limiter le temps système des messages permanents pour l'unité centrale et l'archivage. Dans ce cas, vous pouvez spécifier **msg_persistence=n**.

Les programmes Q Capture et Q Apply enregistrent toujours des messages permanents sur leurs files d'attente d'administration. Q Capture enregistre toujours un message permanent sur sa file d'attente de redémarrage. Par conséquent, la consignation des messages sur ces files d'attente n'est pas affectée par la définition de **msg_persistence**.

Si vous avez créé une file d'attente d'envoi avec l'option DEFPSIST(Y) de telle sorte que la file d'attente transporte les messages permanents par défaut, vous pouvez toujours spécifier **msg_persistence=n** afin que Q Capture envoie des messages non permanents ce qui remplacera la valeur par défaut de la file d'attente.

Si vous choisissez des messages non permanents et que des messages de données sont perdus en raison d'un incident obligeant le gestionnaire de files d'attente à redémarrer, vous devrez effectuer l'une des opérations suivantes pour conserver une synchronisation entre les tables source et cible :

- Arrêtez et démarrez le programme Q Capture et spécifiez un emplacement connu du journal de reprise afin que Q Capture procède à une nouvelle lecture du journal à un emplacement situé avant la perte des messages.
- Arrêtez et démarrez l'abonnement Q pour demander un nouveau chargement (régénération intégrale) de la table cible.

override_restartq

Valeur par défaut : `override_restartq=n`

Méthode de modification : au démarrage de Q Capture

Le paramètre **override_restartq** indique si le programme Q Capture récupère ses informations de démarrage à chaud dans un ensemble de données ou un fichier et non pas dans un message de redémarrage. Par défaut, Q Capture récupère ses

informations de redémarrage dans le message de redémarrage. Si vous indiquez **override_restartq=y**, Q Capture lit le fichier de redémarrage lorsqu'il démarre pour y déterminer le point de démarrage pour chaque file d'attente d'envoi et partition de données. Le fichier est enregistré dans le répertoire spécifié par le paramètre **capture_path** ou dans le répertoire à partir duquel Q Capture a démarré si rien n'est indiqué pour **capture_path**. Le nom de fichier dépend de la plateforme :

z/OS

```
serveur_capture.schéma_capture.QCAP.QRESTART
```

Linux UNIX Windows

```
instance_db2.serveur_capture.schéma_capture  
.QCAP.QRESTART
```

Pour employer ce paramètre, l'ID utilisateur qui démarre Q Capture doit avoir l'autorisation d'ouvrir et de lire le fichier de redémarrage.

part_hist_limit (Linux, UNIX, Windows)

Valeur par défaut : **part_hist_limit=10080** minutes (sept jours)

Méthode de modification : au démarrage de Q Capture ;

Le paramètre **part_hist_limit** indique combien de temps vous voulez que les anciennes données restent dans la table IBMQREP_PART_HIST avant de pouvoir faire l'objet d'un élagage. Ce paramètre contrôle également jusqu'où vous pouvez remonter dans le journal pour redémarrer le programme car Q Capture utilise IBMQREP_PART_HIST pour déterminer les archives de journal à lire pour une table source partitionnée.

Le programme Q Capture utilise l'historique des partitions stocké dans la table IBMQREP_PART_HIST pour faciliter la gestion des changements de partition comme ajouter, joindre ou déconnecter. Une ligne, identifiée par un numéro d'ordre de journal (LSN), est insérée pour chaque partition dans la table source en deux occasions :

- Les premiers abonnements Q ou membre d'un ensemble d'abonnements de la table sont activés.
- La table est modifiée pour ajouter, joindre ou déconnecter une partition.

Si vous répliquez des tables partitionnées sous Linux, UNIX ou Windows, assurez-vous que la valeur **part_hist_limit** est suffisante pour permettre le réenregistrement possible des enregistrements de journal passés dans le cas d'une défaillance du système ou d'une autre indisponibilité.

prune_interval

Valeur par défaut : **prune_interval=300** secondes (5 minutes)

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **prune_interval** détermine la fréquence de recherche de lignes candidates à un élagage par le programme Q Capture dans les tables

IBMQREP_CAPMON, IBMQREP_CAPQMON, IBMQREP_SIGNAL et IBMQREP_CAPTRACE. Par défaut, cette inspection intervient toutes les 300 secondes (5 minutes).

Votre fréquence d'élagage dépend de la rapidité de croissance de vos tables de contrôle et de ce que vous comptez en faire :

Réduisez l'intervalle d'élagage pour gérer des tables de contrôle

Un intervalle d'élagage plus court peut s'avérer nécessaire si les tables IBMQREP_CAPMON et IBMQREP_CAPQMON croissent trop rapidement en raison d'un intervalle de contrôle écourté. Si ces tables de contrôle, comme d'autres, ne sont pas assez souvent élaguées, elles peuvent excéder leurs limites d'espace table et provoquer alors l'arrêt du programme Q Capture. Par contre si elles sont élaguées trop souvent ou au cours de périodes de pointe, cette opération peut faire interférence avec d'autres programmes s'exécutant sur le même système.

Augmentez l'intervalle d'élagage pour conservation des enregistrements

Vous pouvez conserver un historique plus lointain des performances d'un programme Q Capture en élaguant moins fréquemment la table IBMQREP_CAPTRACE et les autres.

L'intervalle d'élagage opère de pair avec les paramètres **trace_limit**, **monitor_limit**, et **signal_limit**, qui déterminent quand les données sont suffisamment anciennes pour élagage. Par exemple, si l'intervalle **prune_interval** est de 300 secondes et la valeur de **trace_limit** de 10 080 secondes, un programme Q Capture tentera un élagage toutes les 300 secondes. Si le programme Q Capture détecte dans la table IBMQREP_CAPTRACE des lignes datant de plus de 10 080 minutes (7 jours), elles sont élaguées.

pwdfile

Valeur par défaut : `pwdfile=capture_path/asnpwd.aut`

Méthode de modification : au démarrage de Q Capture

Le paramètre **pwdfile** indique le nom du fichier de mot de passe qui permet de se connecter aux bases de données à partitions multiples. Si vous n'utilisez pas le paramètre **pwdfile** pour indiquer le nom d'un fichier de mot de passe, le programme Q Capture recherche un fichier avec le nom `asnpwd.aut` dans le répertoire indiqué par le paramètre **capture_path**. Si aucun paramètre **capture_path** n'est défini, cette commande recherche le fichier des mots de passe dans le répertoire duquel la commande a été appelée.

Vous pouvez créer un fichier de mot de passe avec la commande **asnpwd**. Utilisez l'exemple suivant pour créer un fichier de mot de passe avec le nom par défaut `asnpwd.aut` dans le répertoire de travail : `asnpwd INIT`.

qfull_num_retries

Valeur par défaut : `qfull_num_retries=30`

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Vous pouvez utiliser le paramètre **qfull_num_retries** pour indiquer le nombre de fois que le programme Q Capture essaie de mettre un message en file d'attente

lorsque l'opération MQPUT initiale échoue en raison de la saturation de la file d'attente. La valeur par défaut est 30 relances et le nombre maximum est 1000 relances. Une valeur de 0 donne l'instruction au programme Q Capture de s'arrêter dès qu'une opération MQPUT échoue. La syntaxe de remplacement pour le paramètre **qfull_num_retries** est **RN**.

Par exemple, la commande suivante spécifie qu'un programme Q Capture relance l'opération MQPUT 100 fois avant de s'arrêter :

```
asnqcap capture_server=SAMPLE capture_schema=ASN1 qfull_num_retries=100
```

qfull_retry_delay

Valeur par défaut : **qfull_retry_delay**=250 millisecondes

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Vous pouvez utiliser le paramètre **qfull_retry_delay** pour indiquer la durée, en millisecondes, de l'attente du programme Q Capture entre les tentatives MQPUT lorsque l'opération MQPUT initiale échoue en raison de la saturation de la file d'attente. La plage de valeurs autorisée s'étend de 10 millisecondes à 3600000 millisecondes (1 heure). Le délai d'attente par défaut est de 250 millisecondes ou la valeur du paramètre **commit_interval** si cette valeur est inférieure. (La valeur par défaut de **commit_interval** est 500 millisecondes.) Si la valeur indiquée est supérieure à celle de **commit_interval**, elle sera définie sur une valeur inférieure à la valeur de **commit_interval**. La syntaxe de remplacement pour le paramètre **qfull_retry_delay** est **RD**.

Par exemple, la commande suivante spécifie qu'un programme Q Capture relance l'opération MQPUT 50 fois avant de s'arrêter, avec un délai d'attente de 10000 millisecondes (10 secondes) :

```
asnqcap capture_server=SAMPLE capture_schema=ASN1 qfull_num_retries=50  
qfull_retry_delay=10000
```

signal_limit

Valeur par défaut : **signal_limit**=10 080 minutes (7 jours)

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **signal_limit** spécifie combien de temps les lignes doivent demeurer dans la table IBMQREP_SIGNAL avant de pouvoir être élaguées.

Par défaut, à chaque intervalle d'élagage, le programme Q Capture efface toutes les lignes datant de plus de 10 080 minutes (7 jours).

La table IBMQREP_SIGNAL contient des signaux insérés par l'utilisateur ou par une application utilisateur. Elle comporte également des signaux correspondants insérés par le programme Q Capture après réception de messages de contrôle provenant du programme Q Apply ou d'une application utilisateur. Le programme Q Capture détecte ces signaux lors de la lecture dans la table IBMQREP_SIGNAL de l'enregistrement du journal pour insertion.

Ces signaux indiquent au programme Q Capture d'arrêter son exécution, d'activer ou de désactiver un abonnement Q ou une publication, d'ignorer une transaction

de base de données dans le journal ou d'invalider une file d'attente d'envoi. De plus, le signal LOADDONE indique au programme Q Capture le chargement d'une table cible.

Vous pouvez ajuster cette limite en fonction de votre environnement. Réduisez cette limite pour gérer la taille de la table IBMQREP_SIGNAL. Pour la réplication Q bidirectionnelle, le programme Q Apply insère un signal dans la table IBMQREP_SIGNAL pour chaque transaction qu'il reçoit et applique afin de garantir que le programme Q Capture ne recapture pas la transaction. Si vous utilisez un grand nombre d'abonnements Q bidirectionnels, cette table peut attendre des proportions imposantes, auquel cas vous pouvez envisager de réduire la limite de signal par défaut afin de pouvoir l'élaguer plus fréquemment. Étendez cette limite pour utiliser cette table à des fins de conservation des enregistrements.

sleep_interval

Valeurs par défaut : **sleep_interval**=500 millisecondes (0,5 seconde) pour les sources DB2 ; 2000 millisecondes (2 secondes) pour les sources Oracle

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **sleep_interval** spécifie le délai d'attente, en millisecondes, du programme Q Capture après avoir atteint la fin du journal actif et assemblé les transactions éventuelles encore en mémoire.

Par défaut, le programme Q Capture demeure inactif pendant 5 000 millisecondes (5 secondes). Après ce délai, le programme reprend la lecture du journal. Vous pouvez régler cet intervalle d'inactivité en fonction de votre environnement :

Raccourcissez cet intervalle si vous voulez réduire les temps d'attente

Un intervalle plus court peut améliorer les performances en réduisant le temps d'attente (temps pris par une transaction pour transiter de la source à la cible), en réduisant les périodes d'inactivité et en augmentant le débit dans un environnement avec volume élevé de transactions.

Augmentez l'intervalle d'inactivité si vous désirez conserver vos ressources

Un intervalle d'inactivité plus long permet une économie potentielle des ressources de l'unité centrale dans un environnement caractérisé par un trafic peu important sur la base de données source ou avec des cibles ne requérant pas de mises à jour fréquentes.

stale

Valeur par défaut : **stale**=3600

Méthode de modification : au démarrage de Q Capture ;

Le paramètre **stale** indique le temps en secondes pendant lequel le programme Q Capture attend pour émettre un message d'avertissement ou faire toute autre action après avoir détecté une transaction à exécution longue sans enregistrement de journal de validation ou d'annulation. Le comportement du programme dépend de la plateforme de la source. Sous z/OS, Q Capture émet des messages d'avertissement s'il n'a pas vu d'enregistrement validé ou annulé pendant une heure (**stale**=3600). Sous z/OS et Linux, UNIX, et Windows, Si une transaction est exécutée pendant le nombre de secondes indiqué par le paramètre **stale** et que le programme Q Capture ne trouve aucune opération de ligne dans le journal pour

cette transaction, il émet des messages d'avertissement et incrémente le numéro d'ordre de journal qu'il considère comme étant la transaction "en vol" la plus ancienne non validée ou annulée. Si certaines lignes ont été capturées pour la transaction, seul le message d'avertissement est émis.

startallq

Valeur par défaut : `startallq=y`

Méthodes de modification : au démarrage de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **startallq** indique si le programme Q Capture active toutes les files d'attente d'envoi au cours du démarrage. Vous pouvez utiliser ce paramètre pour garder une file d'attente d'envoi désactivée inactive.

Par défaut, lorsque le programme Q Capture démarre, il active toutes les files d'attente d'envoi qui ne sont pas déjà actives (A). Si vous indiquez **startallq=n**, lorsque le programme Q Capture démarre, il n'active pas toutes les files d'attente d'envoi inactives (I).

startmode

Valeur par défaut : `startmode=warmsi`

Méthodes de modification : au démarrage de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **startmode** spécifie les étapes à accomplir par un programme Q Capture à son démarrage. Le programme démarre sous l'un des deux modes : à chaud ou à froid. Avec le démarrage à chaud, le programme Q Capture reprend la capture des modifications à partir de l'endroit où il s'était arrêté à sa dernière exécution (on distingue trois types de démarrage à chaud). Si vous choisissez le démarrage à froid, le programme commence sa lecture à la fin du journal. Sélectionnez parmi ces quatre modes de démarrage celui le mieux adapté à votre environnement :

cold Le programme Q Capture vide les files d'attente de redémarrage et d'administration, et commence le traitement de tous les abonnements Q ou publications dont l'état est N (nouveau) ou A (actif). Avec un démarrage à froid, le programme Q Capture commence la lecture du journal de reprise DB2 ou du journal redo Oracle à la fin.

En règle générale, vous ne devriez utiliser un démarrage à froid qu'au tout premier lancement d'un programme Q Capture. Warmsi est le mode de démarrage recommandé. Vous pouvez néanmoins utiliser un démarrage à froid bien qu'il ne s'agisse pas du premier lancement du programme Q Capture si vous souhaitez capturer les modifications à partir de la fin du journal actif et non pas à partir du dernier point de redémarrage. Lors d'un démarrage à froid, Q Capture arrête, puis redémarre les abonnements Q unidirectionnels indiquant une phase de chargements. Concernant la réplication bidirectionnelle et entre homologues, Q Capture arrête uniquement les abonnements Q lors d'un démarrage à froid. Pour démarrer ces abonnements Q, vous devez utiliser les outils d'administration de la réplication ou insérer un signal CAPSTART pour chaque abonnement Q que vous souhaitez démarrer.

Important : Pour éviter des démarrages à froid indésirables, assurez-vous que ce mode de démarrage n'a pas été spécifié dans la table IBMQREP_CAPPARMS.

warmsi (démarrage à chaud ; bascule la toute première fois sur un démarrage à froid) Le programme Q commence sa lecture du journal à partir du point où il s'était arrêté, sauf s'il s'agit de son premier démarrage. Dans ce cas, le programme Q Capture effectue un démarrage à froid. Le mode **warmsi** assure que le programme Q Capture n'utilise un démarrage à froid qu'à son lancement initial.

warmns (démarrage à chaud ; ne bascule jamais sur un démarrage à froid) Le programme Q Capture commence la lecture du journal à l'emplacement où il s'était arrêté. S'il ne peut pas être démarré à chaud, il ne passe pas à un démarrage à froid. Utilisez ce mode de démarrage pour empêcher un programme Q Capture de démarrer à froid à l'improviste. Ce mode de démarrage vous permet de corriger des problèmes (comme une indisponibilité des bases de données ou des espaces table) empêchant le démarrage à chaud. Sous le mode **warmns**, si le programme Q Capture ne parvient pas à exécuter un démarrage à chaud, il s'arrête sans aucune modification des tables.

Lors des démarrages à chaud, le programme Q Capture ne charge que les abonnements Q ou les publications dont l'état n'est pas I (inactif).

term

Valeur par défaut : **term=y**

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **term** détermine si un programme Q Capture continue de s'exécuter lorsque DB2 ou le gestionnaire de files d'attente est indisponible.

Par défaut (**term=y**), le programme Q Capture s'arrête lorsque DB2 ou le gestionnaire de files d'attente n'est pas disponible. Vous pouvez modifier la valeur par défaut (**term=n**) si vous souhaitez que le programme Q Capture poursuive son exécution pendant que DB2 ou le gestionnaire de files d'attente est indisponible. Lorsque DB2 ou le gestionnaire de files d'attente est disponible, Q Capture commence à envoyer des transactions là où il s'est arrêté sans qu'il soit nécessaire de redémarrer le programme.

Sources Oracle : La valeur par défaut de **term** est n (le programme Q Capture continue de s'exécuter lorsque la base de données Oracle est indisponible). Le paramétrage de la valeur sur y n'a aucun effet.

Remarque : Quelle que soit la valeur du paramètre **term**, si les canaux émetteurs ou récepteurs WebSphere MQ s'arrêtent, le programme Q Capture continue de s'exécuter car il ne peut pas détecter le statut des canaux. Cette situation provoque l'arrêt de la réplication car les deux gestionnaires de files d'attente ne peuvent pas communiquer. Si la réplication a été arrêtée sans aucun message des programmes de réplication Q, vérifiez le statut des canaux à l'aide de la commande WebSphere MQ DISPLAY CHSTATUS.

trace_limit

Valeur par défaut : `trace_limit=10 080` minutes (7 jours)

Méthodes de modification : au démarrage de Q Capture ; lors de l'exécution de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **trace_limit** spécifie combien de temps les lignes doivent demeurer dans la table IBMQREP_CAPTRACE avant de pouvoir être élaguées.

Le programme Q Capture insère dans la table IBMQREP_CAPTRACE tous les messages d'information, d'avertissement et d'erreur. Par défaut, les lignes de plus de 10 080 minutes (7 jours) sont élaguées à chaque intervalle d'élagage. Modifiez la limite de trace en fonction de vos besoins pour des informations d'audit.

trans_batch_sz

Valeur par défaut : `trans_batch_sz=1`

Méthodes de modification : au démarrage de Q Capture ;

Le paramètre **trans_batch_sz** indique le nombre de transactions de base de données source que Q Capture regroupe dans un message WebSphere MQ. Vous pouvez utiliser ce paramètre pour réduire la consommation d'unité centrale à la source lorsque la charge de travail de la réplication consiste généralement en petites transactions, par exemple d'une ou deux lignes.

La valeur par défaut est 1 (pas de lot) et la taille de lot maximale est de 128. Q Capture respecte la valeur du paramètre **trans_batch_sz**, sauf si l'intervalle de validation de Q Capture est atteint avant que le nombre de transactions atteigne cette valeur ou si la taille totale des transactions dans le lot dépasse la valeur du paramètre **max_message_size** pour la file d'attente d'envoi ou la valeur de longueur maximale de message WebSphere MQ (MAXMSGL) pour la file d'attente d'envoi.

Remarque : L'utilisation de ce paramètre avec d'autres paramètres ou cibles CDD peut engendrer des résultats indésirables. Pour plus de détails, voir `trans_batch_sz` parameter.

warnlogapi (z/OS)

Valeur par défaut : `warnlogapi=0`

Méthodes de modification : au démarrage de Q Capture ; table IBMQREP_CAPPARMS

Le paramètre **warnlogapi** indique le temps en millisecondes pendant lequel le programme Q Capture attend l'interface IFI (instrumentation facility interface) DB2 for z/OS ou l'interface de programme d'application de lecture de journal DB2 for Linux, UNIX et Windows pour renvoyer les enregistrements de journal avant que Q Capture affiche un avertissement vers la sortie standard.

warntxsz

Valeur par défaut : `warntxsz=0` Mo

Méthodes de modification : au démarrage de Q Capture

Vous pouvez utiliser le paramètre **warntxsz** pour détecter les transactions très volumineuses. Ce paramètre demande au programme Q Capture d'envoyer un message d'avertissement lorsqu'il rencontre des transactions plus volumineuses que la taille indiquée. Vous fournissez une valeur seuil en mégaoctets et les transactions qui dépassent le seuil provoquent l'affichage d'un message d'avertissement. Q Capture émet plusieurs messages d'avertissement si la taille de transaction est un multiple de la valeur de **warntxsz**. Par exemple, si vous définissez **warntxsz** sur 10 Mo et que Q Capture rencontre une transaction de 30 Mo, trois avertissements sont envoyés (un pour 10 Mo, un pour 20 Mo et un pour 30 Mo).

Q Capture émet le message ASN0664W lorsque la taille d'une transaction dépasse la valeur de **warntxsz**. La valeur par défaut de 0 Mo signifie qu'aucun avertissement n'est jamais envoyé.

Exemples d'utilisation de la commande **asnqcap**

Ces exemples illustrent comment utiliser la commande **asnqcap**.

Remplacement du mode de démarrage et des paramètres de journal par défaut

Pour démarrer un programme Q Capture sur un serveur appelé sourcedb avec un schéma d'alpha, pour démarrer le programme en utilisant le mode de démarrage à froid et pour remplacer provisoirement les paramètres par défaut pour **logreuse** et **logstdout**, émettez la commande suivante :

```
asnqcap capture_server=sourcedb capture_schema="alpha" startmode=cold
logreuse=y logstdout=y
```

Remplacement de l'intervalle de validation par défaut

Pour démarrer un programme Q Capture et lui ordonner de valider des messages dans les files d'attente à une fréquence plus élevée que celle par défaut, émettez la commande suivante :

```
asnqcap capture_server=sourcedb capture_schema="alpha" commit_interval=250
```

Modification de l'allocation de mémoire

Pour démarrer un programme Q Capture et augmenter provisoirement la quantité de mémoire par défaut qu'il utilise pour générer des transactions, émettez la commande suivante :

```
asnqcap capture_server=sourcedb capture_schema="alpha" memory_limit=64
```

Spécification de l'emplacement du journal et du fichier de travail

Pour démarrer un programme Q Capture et envoyer ses fichiers de travail vers le répertoire /home/files/qcapture, émettez la commande suivante :

```
asnqcap capture_server=sourcedb capture_schema="alpha"
capture_path="/home/files/qcapture"
```

Spécification de **lsn** et **maxcmtseq**

Pour démarrer un programme Q Capture sur un serveur nommé testdb, et indiquer qu'après un démarrage à chaud le programme démarre à un numéro de

modification système (SCN) (spécifié par le paramètre **lsn**) de 0000:0000:0000:115b:7704 et **maxcmtseq** of 41c2:2264:0000:0004:0000, émettez la commande suivante :

```
asnqcap capture_server=testdb lsn=0000:0000:0000:115b:7704
maxcmtseq=41c2:2264:0000:0004:0000:
```

Démarrage à partir de la fin du journal

Pour démarrer un programme Q Capture à partir de la fin du journal sans déclencher un chargement (régénération intégrale) de la table cible, émettez la commande suivante :

```
asnqcap capture_server=testdb lsn=FFFF:FFFF:FFFF:FFFF:FFFF
maxcmtseq=FFFF:FFFF:FFFF:FFFF:FFFF
```

Ignorer une transaction spécifique

Pour démarrer un programme Q Capture en partage de données sur z/OS et indiquer que le programme ignore une transaction dont l'ID de transaction est BD71:1E23:B089 (l'ID membre pour le partage de données est 0001), exécutez la commande suivante :

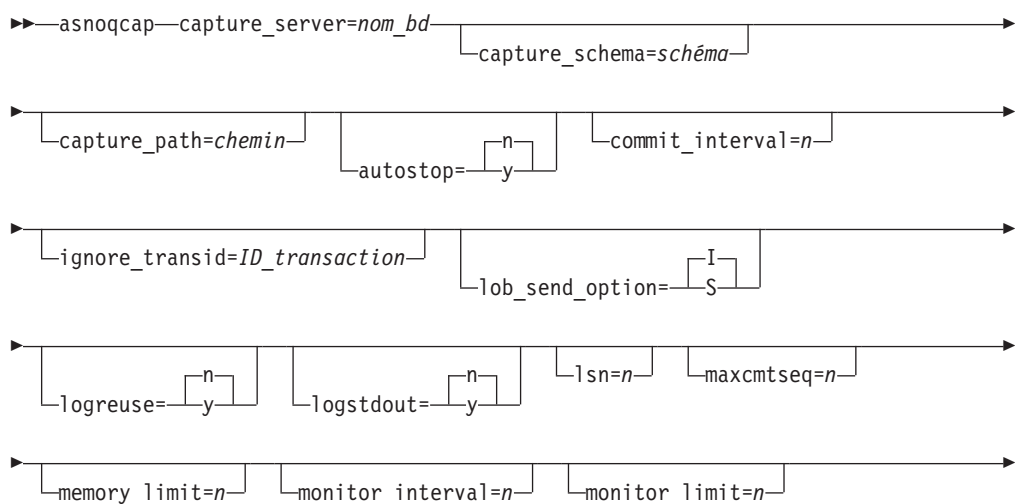
```
asnqcap capture_server=sample capture_schema=ASN
ignore_transid=0000:BD71:1E23:B089:0001
```

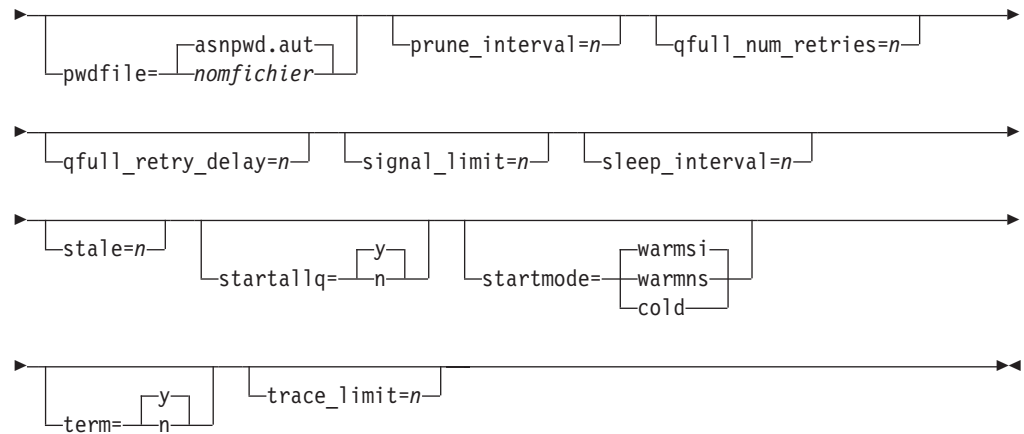
asnqcap : Démarrage d'un programme Q Capture pour une base de données Oracle

Utilisez la commande **asnqcap** pour démarrer un programme Q Capture pour des bases de données Oracle sur les systèmes Linux, UNIX, ou Windows. Exécutez cette commande à l'invite du système d'exploitation ou dans un script de shell. Les paramètres de démarrage que vous définissez ne s'appliqueront qu'à cette session.

Une fois que vous avez démarré le programme Q Capture, il s'exécute continuellement jusqu'à ce que vous l'arrêtez ou qu'il détecte une erreur irréparable.

Syntaxe





Paramètres

Le tableau suivant définit les paramètres d'appel pour la commande **asnoqcap**.

Tableau 39. Définitions des paramètres d'appel *asnoqcap*

Paramètre	Définition
capture_server=nombd	Indique le nom de la base de données Oracle qui contient les tables de contrôle Q Capture. Si vous n'indiquez pas de serveur Q Capture, ce paramètre aura comme valeur par défaut la valeur de la variable d'environnement ORACLE_SID.
capture_schema=schéma	Définit un nom identifiant le programme Q Capture que vous voulez démarrer. Le schéma par défaut est ASN.
capture_path=chemin	Définit l'emplacement où vous voulez qu'un programme Q Capture écrive son journal et ses fichiers de travail. La valeur par défaut est le répertoire dans lequel vous avez appelé la commande asnoqcap . Cet emplacement est un nom de chemin d'accès absolu, et vous devez le mettre entre guillemets et conserver la casse.
autostop=y/n	Indique si un programme Q Capture s'arrête après avoir atteint la fin du journal redo Oracle actif. y Le programme Q Capture s'arrête lorsqu'il atteint la fin du journal redo Oracle actif. n (par défaut) Le programme Q Capture ne s'arrête pas après avoir atteint la fin du journal redo Oracle actif.
commit_interval=n	Définit la fréquence, en millisecondes, à laquelle un programme Q Capture émet un appel MQCMIT. Cet appel signale au gestionnaire de file d'attente WebSphere MQ qu'il doit rendre les messages de données et d'information qui ont été placés en attente accessibles au programme Q Apply ou à une application d'abonnement. La valeur par défaut est de 1000 millisecondes (1 seconde).

Tableau 39. Définitions des paramètres d'appel *asnoqcap* (suite)

Paramètre	Définition
ignore_transid= <i>ID_transaction</i>	<p>Indique que le programme Q Capture ignorera la transaction identifiée par <i>ID_transaction</i>. Les transactions ne sont pas répliquées ni publiées.</p> <p>L'ID de transaction est la valeur XID de la vue V\$CONTENTS de la fonctionnalité Oracle LogMiner. L'ID est une valeur RAW(8). Lorsqu'il est affiché dans le texte, l'ID de transaction est formaté dans une chaîne hexadécimale (16 chiffres au total).</p> <p>Conseil : La version raccourcie transid est également acceptable pour ce paramètre.</p>
lob_send_option= I/S	<p>Indique si le programme Q Capture envoie des valeurs LOB intégrées (I) au sein du message de transaction, ou s'il envoie des valeurs LOB dans des messages séparés (S). Par défaut, les valeurs LOB sont envoyées au sein du message de transaction.</p>
logreuse= y/n	<p>Définit si un programme Q Capture réutilise ou ajoute des messages à son fichier journal de diagnostic. Le nom du fichier journal est <i>serveur_capture.schema_capture.QCAP.log</i>.</p> <p>y Au redémarrage, le programme Q Capture réutilise le fichier journal en nettoyant le fichier puis en écrivant dans le fichier vierge.</p> <p>n (par défaut) Le programme Q Capture ajoute des messages au fichier journal, même après le redémarrage du programme Q Capture.</p>
logstdout= y/n	<p>Définit si un programme Q Capture envoie des messages de journal à son fichier journal de diagnostic et à la console.</p> <p>y Le programme Q Capture envoie des messages de journal à son fichier journal et à la console (stdout).</p> <p>n (par défaut) Le programme Q Capture dirige la plupart des messages de journal vers le fichier journal uniquement.</p> <p>Les messages d'activation et de désactivation de l'initialisation, de l'arrêt et de l'abonnement vont vers la console (stdout) et vers le fichier journal, quelle que soit la valeur de ce paramètre.</p>

Tableau 39. Définitions des paramètres d'appel asnoqcap (suite)

Paramètre	Définition
l_{sn}=n	<p>Indique l'enregistrement de journal à partir duquel le programme Q Capture commence à lire pendant un redémarrage à chaud. Cette valeur représente le premier enregistrement de journal que le programme Q Capture a identifié pour lequel aucun enregistrement de validation ou d'abandon n'a encore été trouvé.</p> <p>Vous pouvez trouver la valeur du paramètre l_{sn} dans les messages ASN7108I et ASN7109I : Par exemple, dans le message suivant, la valeur de l_{sn} est le "numéro d'ordre de journal le plus bas d'une transaction à valider" ou 0000:0000:0F1B:8629:0000:</p> <pre>2010-03-08-13.34.36.913067 ASN7108I "Q Capture" : "ASN3" : "WorkerThread" : A l'initialisation du programme, le numéro d'ordre de journal le plus élevé d'une transaction traitée avec succès est "0000:0000:0F1B:8486:0000" et le numéro d'ordre de journal le plus bas d'une transaction à valider est "0000:0000:0F1B:8629:0000".</pre> <p>La valeur l_{sn} se compose de cinq ensembles de quatre caractères hexadécimaux dont le premier ensemble représente le numéro de changement de système Oracle (SCN) et le dernier ensemble un numéro d'ordre généré par Q Capture. Par exemple, la valeur l_{sn} suivante combine la version hexadécimale de SCN 123456 (1E240) avec un numéro d'ordre de 0000 :</p> <p>Valeur hexadécimale de SCN numéro d'ordre 123456 0000:0000:0001:E240: 0000</p> <p>Q Capture ignore les zéros de tête et les deux points(:). Vous pouvez aussi spécifier la valeur l_{sn} ci-dessus des manières suivantes :</p> <pre>l_{sn}=1E2400000 l_{sn}=1E240:0000 l_{sn}=1:E240:0000 l_{sn}=000000000001E2400000 l_{sn}=0000:0000:0001:E240:0000</pre> <p>Si vous spécifiez 0 pour les paramètres l_{sn} et maxcmtseq, Q Capture démarre en lisant le journal au SCN en cours. Vous pouvez préciser la valeur l_{sn} à partir du message ASN7108I ou ASN7109I pour l_{sn} et maxcmtseq, ou spécifier la valeur à partir du message pour l_{sn} et maxcmtseq=0, et Q Capture démarre en lisant au numéro d'ordre de journal donné.</p> <p>Spécifiez l_{sn}=FFFF:FFFF:FFFF:FFFF:FFFF pour démarrer à partir de la fin du journal sans déclencher de chargement (régénération intégrale) de la table cible.</p>

Tableau 39. Définitions des paramètres d'appel asnoqcap (suite)

Paramètre	Définition
maxcmtseq = <i>n</i>	<p>Indique l'enregistrement de journal le plus récent d'une transaction validée avec succès que le programme Q Capture lit pendant sa dernière exécution. Vous pouvez utiliser la valeur maxcmtseq avec la valeur pour le paramètre lsn pour démarrer Q Capture depuis un point connu du journal.</p> <p>Vous pouvez trouver cette valeur dans les messages ASN7108I et ASN7109I. Par exemple, dans le message suivant, la valeur de maxcmtseq est le "numéro d'ordre de journal le plus élevé d'une transaction correctement traitée" ou 0000:0000:0F1B:8486:0000:</p> <pre>2010-03-08-13.34.36.913067 ASN7108I "Q Capture" : "ASN3" : "WorkerThread" : A l'initialisation du programme, le numéro d'ordre de journal le plus élevé d'une transaction traitée avec succès est "0000:0000:0F1B:8486:0000" et le numéro d'ordre de journal le plus bas d'une transaction à valider est "0000:0000:0F1B:8629:0000".</pre> <p>Le format de la valeur maxcmtseq est le même que celui de lsn (voir ci-dessus).</p> <p>Si vous créez la valeur maxcmtseq manuellement, vous pouvez spécifier le numéro d'ordre comme égal à 0000 car Q Capture démarre en lisant le premier enregistrement de journal d'un SCN donné. Si vous utilisez la valeur dans les messages ASN7108I et ASN7109I, le numéro d'ordre peut être différent de 0000.</p> <p>Si vous spécifiez 0 pour les paramètres lsn et maxcmtseq, Q Capture démarre en lisant le journal au SCN en cours.</p>
memory_limit = <i>n</i>	<p>Définit la quantité de mémoire, en mégaoctets, qu'un programme Q Capture peut utiliser pour générer des transactions. Après l'utilisation de cette attribution, les transactions en mémoire se déversent dans un fichier. La valeur par défaut est 500 Mo.</p> <p>La valeur maximale autorisée pour ce paramètre est de 100 Go.</p>
msg_persistence = <i>y/n</i>	<p>Indique si un programme Q Capture écrit des messages permanents (consignés) dans des files d'attente WebSphere MQ.</p> <p>y (par défaut) Q Capture enregistre des messages permanents sur toutes les files d'attente. Les messages sont consignés par le gestionnaire de files d'attente et peuvent être récupérés.</p> <p>n Q Capture enregistre des messages non permanents sur toutes les files d'attente. Les messages ne sont pas consignés et ne peuvent pas être récupérés.</p> <p>Remarque : Si vous spécifiez msg_persistence=n, assurez-vous que les files d'attente sont vides avant d'arrêter le gestionnaire de files d'attente en arrêtant tout d'abord Q Capture puis en vérifiant que Q Apply a vidé toutes les files d'attente.</p>

Tableau 39. Définitions des paramètres d'appel *asnoqcap* (suite)

Paramètre	Définition
<code>monitor_interval=n</code>	<p>Définit la fréquence, en millisecondes, à laquelle le programme Q Capture ajoute une ligne aux tables IBMQREP_CAPMON et IBMQREP_CAPQMON. La valeur par défaut est de 30000 millisecondes (30 secondes).</p> <p>Important pour les utilisateurs du tableau de bord de Q Replication : Lorsque cela est possible, vous devez synchroniser le paramètre <code>monitor_interval</code> de Q Capture avec l'intervalle d'actualisation du tableau de bord (à quelle fréquence le tableau de bord récupère des informations de performance des tables de contrôle Q Capture et Q Apply). L'intervalle d'actualisation par défaut du tableau de bord est de 10 secondes (10000 millisecondes). Si la valeur de <code>monitor_interval</code> est supérieure à l'intervalle d'actualisation du tableau de bord, ce dernier s'actualise alors qu'aucune nouvelle donnée de surveillance n'est disponible.</p>
<code>monitor_limit=n</code>	<p>Définit le nombre de minutes pendant lesquelles les lignes restent dans les tables IBMQREP_CAPMON et IBMQREP_CAPQMON avant de pouvoir être élaguées. A chaque intervalle d'élagage, le programme Q Capture élague les lignes de ces tables qui sont plus anciennes que cette limite en fonction de l'horodatage actuel. La valeur par défaut est 10080 minutes (sept jours).</p>
<code>pwdfile=nomfichier</code>	<p>Définit le nom du fichier des mots de passe utilisé pour se connecter à plusieurs bases de données de partition. Si vous ne spécifiez pas de mot de passe, la valeur par défaut est <code>asnpwd.aut</code>.</p> <p>Cette commande recherche le fichier des mots de passe dans le répertoire défini par le paramètre <code>capture_path</code>. Si aucun paramètre <code>capture_path</code> n'est défini, cette commande recherche le fichier des mots de passe dans le répertoire où la commande a été appelée.</p> <p><code>asnpwd</code> : Création et conservation des fichiers de mots de passe décrit comment créer et modifier un fichier de mots de passe.</p>
<code>prune_interval=n</code>	<p>Définit la fréquence, en secondes, à laquelle un programme Q Capture recherche dans les tables IBMQREP_SIGNAL, IBMQREP_CAPTRACE, IBMQREP_CAPMON et IBMQREP_CAPQMON des lignes suffisamment anciennes pour être élaguées. La valeur par défaut est 300 secondes (cinq minutes).</p>
<code>qfull_num_retries=n</code>	<p>Spécifie le nombre de relances à essayer. La valeur par défaut est 30 relances et le nombre maximum est 1000 relances. Une valeur de 0 donne l'instruction au programme Q Capture de s'arrêter dès qu'une opération MQPUT échoue. La syntaxe de remplacement pour le paramètre <code>qfull_num_retries</code> est RN.</p>
<code>qfull_retry_delay=n</code>	<p>Spécifie le temps en millisecondes que le programme Q Capture attend entre les tentatives MQPUT. La plage de valeurs autorisée s'étend de 10 millisecondes à 3600000 millisecondes (1 heure). Le délai d'attente par défaut est de 250 millisecondes ou la valeur du paramètre <code>commit_interval</code> si cette valeur est inférieure. (La valeur par défaut de <code>commit_interval</code> est 500 millisecondes.) La syntaxe de remplacement pour le paramètre <code>qfull_retry_delay</code> est RD.</p>

Tableau 39. Définitions des paramètres d'appel *asnoqcap* (suite)

Paramètre	Définition
signal_limit = <i>n</i>	Définit le nombre de minutes pendant lesquelles les lignes restent dans la table IBMQREP_SIGNAL avant de pouvoir être élaguées. A chaque intervalle d'élagage, le programme Q Capture élague les lignes de la table IBMQREP_SIGNAL qui sont plus anciennes que la limite de signal en fonction de l'horodatage actuel. La valeur par défaut est 10080 minutes (sept jours).
sleep_interval = <i>n</i>	Indique le nombre de millisecondes durant lesquelles un programme Q Capture reste en veille après le traitement du journal actif et des transactions restant éventuellement en mémoire. La valeur par défaut est de 2000 millisecondes (2 secondes).
stale = <i>n</i>	Indique le nombre de secondes pendant lesquelles le programme Q Capture attend pour émettre un message d'avertissement ou faire toute autre action après avoir détecté une transaction à exécution longue sans enregistrement de journal de validation ou d'annulation. Le comportement du programme dépend de la plateforme de la source. Si une transaction est exécutée pendant le nombre de secondes indiqué par le paramètre stale et que le programme Q Capture ne trouve aucune opération de ligne dans le journal pour cette transaction, il émet un message d'avertissement, ne réplique pas la transaction et incrémente le numéro d'ordre de journal qu'il considère comme étant la transaction en cours la plus ancienne non validée ou annulée. Si certaines lignes ont été capturées pour la transaction, seul le message d'avertissement est émis.
startallq = <i>y/n</i>	Définit si le programme Q Capture active toutes les files d'attente d'envoi au cours du démarrage. Vous pouvez utiliser ce paramètre pour garder une file d'attente d'envoi désactivée inactive. y (par défaut) Lorsque le programme Q Capture démarre, il active toutes les files d'attente d'envoi qui ne sont pas déjà actives (A). n Lorsque le programme Q Capture démarre, il n'active pas toutes les files d'attente d'envoi qui sont inactives (I).

Tableau 39. Définitions des paramètres d'appel *asnoqcap* (suite)

Paramètre	Définition
startmode=mode	Définit les opérations effectuées par le programme Q Capture au démarrage.
	<p>warmsi (par défaut)</p> <p>Le programme Q Capture commence la lecture du journal à l'endroit où il s'était arrêté, sauf s'il est démarré pour la première fois. Lorsque vous démarrez le programme Q Capture pour la première fois, celui-ci effectue un démarrage à froid. Le mode de démarrage warmsi garantit que le programme Q Capture démarre à froid uniquement lors de son démarrage initial.</p> <p>warmns</p> <p>Le programme Q Capture commence la lecture du journal à l'emplacement où il s'était arrêté. Si le programme Q Capture ne peut pas être démarré à chaud, il ne passe pas à un démarrage à froid. Le mode de démarrage warmns empêche le programme Q Capture de démarrer à froid de façon intempestive. Dans le cas d'un démarrage à chaud, le programme Q Capture reprend le traitement où il s'était arrêté. Si des erreurs se produisent après le démarrage du programme Q Capture, le programme s'arrête sans modifier aucune table.</p> <p>cold</p> <p>Le programme Q Capture vide les files d'attente de redémarrage et d'administration, et commence le traitement de tous les abonnements Q ou publications dont l'état est N (nouveau) ou A (actif). Avec un démarrage à froid, le programme Q Capture commence la lecture à la fin du journal redo Oracle actif.</p> <p>Pendant les démarrages à chaud, le programme Q Capture ne chargera que les abonnements Q ou les publications dont l'état n'est pas I (inactif).</p>
term=o/n	Spécifie si le programme Q Capture se termine si la base de données source est mise au repos. Le paramètre term n'a pas d'effet sur les sources Oracle. La valeur par défaut est N, ce qui indique que le programme Capture continue de s'exécuter si Oracle est mis au repos ou arrêté. Lorsqu'Oracle est sorti du mode de repos, le programme Q Capture continue le traitement à l'endroit où il s'était arrêté lorsqu'Oracle a été mis au repos. Si la source Oracle est arrêtée et redémarrée, le programme Q Capture doit être arrêté et redémarré.
trace_limit=n	Indique le nombre de minutes pendant lequel les lignes doivent rester dans la table IBMQREP_CAPTRACE avant de pouvoir être élaguées. A chaque intervalle d'élagage, le programme Q Capture élague les lignes de cette table qui sont plus anciennes que la limite de traçage en fonction de l'horodatage actuel. La valeur par défaut est 10080 minutes (sept jours).

Exemples pour *asnoqcap*

Les exemples suivants illustrent comment utiliser la commande **asnoqcap**.

Exemple - Remplacement du mode de démarrage et des paramètres de journal par défaut

Pour démarrer un programme Q Capture sur un serveur appelé sourcedb avec un schéma d'alpha utilisant le mode de démarrage à froid et pour remplacer provisoirement les paramètres par défaut pour **logreuse** et **logstdout**, émettez la commande suivante :

```
asnoqcap capture_server=sourcedb capture_schema="alpha" startmode=cold
logreuse=y logstdout=y
```

Exemple - Remplacement de l'intervalle de validation de messages par défaut

Pour démarrer un programme Q Capture et lui ordonner de valider des messages dans les files d'attente à une fréquence plus élevée que celle par défaut, émettez la commande suivante :

```
asnoqcap capture_server=sourcedb capture_schema="alpha" commit_interval=250
```

Exemple - Modification de l'allocation de mémoire

Pour démarrer un programme Q Capture et augmenter provisoirement la quantité de mémoire par défaut qu'il utilise pour générer des transactions, émettez la commande suivante :

```
asnoqcap capture_server=sourcedb capture_schema="alpha" memory_limit=64
```

Exemple - Spécification de l'emplacement du journal et du fichier de travail

Pour démarrer un programme Q Capture et envoyer ses fichiers de travail vers le répertoire /home/files/qcapture, émettez la commande suivante :

```
asnoqcap capture_server=sourcedb capture_schema="alpha"
capture_path="/home/files/qcapture"
```

Exemple - Spécification de lsn et maxcmtseq

Pour démarrer un programme Q Capture sur un serveur nommé testdb, et indiquer qu'après un redémarrage à chaud le programme démarre à un numéro de modification système (SCN) Oracle (spécifié par le paramètre **lsn**) de 0000:0000:0F1B:8629:0000 et **maxcmtseq** de 0000:0000:0F1B, émettez la commande suivante :

```
asnoqcap capture_server=testdb lsn=0000:0000:0F1B:8629:0000
maxcmtseq=0000:0000:0F1B:8486:0000
```

Exemple - Démarrage à partir de la fin du journal

Pour démarrer un programme Q Capture à partir de la fin du journal sans déclencher un chargement (régénération intégrale) de la table cible, émettez la commande suivante :

```
asnoqcap capture_server=testdb lsn=FFFF:FFFF:FFFF:FFFF:FFFF
maxcmtseq=FFFF:FFFF:FFFF:FFFF:FFFF
```

Exemple - Ignorer une transaction spécifique

Pour démarrer un programme Q Capture sur un serveur Oracle et spécifier que le programme ignore une transaction avec un ID de transaction BD71:1E23:B089:0001, émettez la commande suivante :

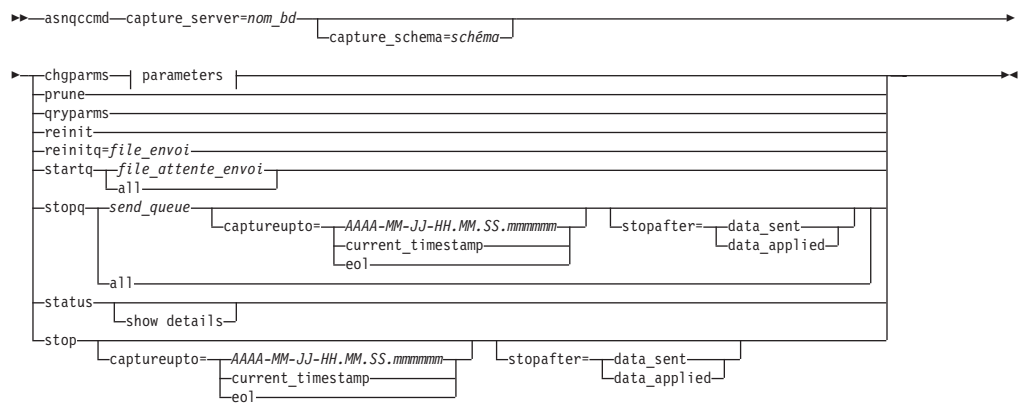
```
asnoqcap capture_server=ora_sample capture_schema=ASN
ignore_transid=BD71:1E23:B089:0001
```

asnlccmd : utilisation d'un programme Q Capture en cours d'exécution

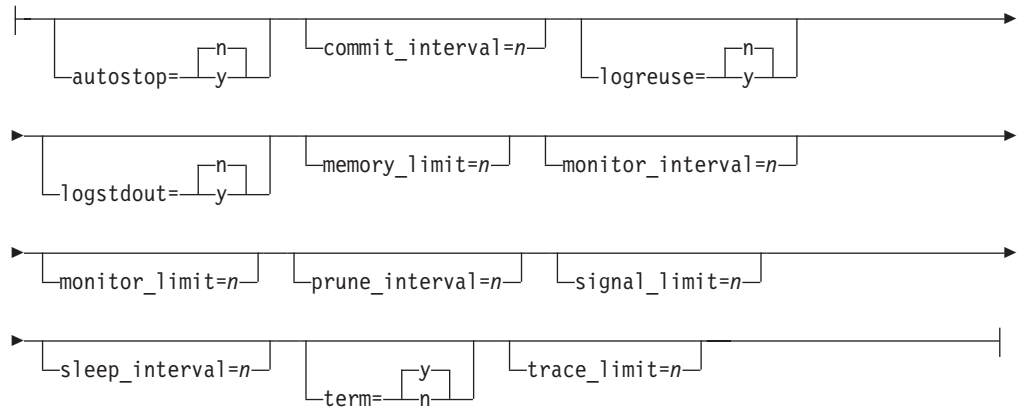
Utilisez **asnlccmd** pour envoyer une commande à un programme Q Capture en cours d'exécution sous Linux, UNIX, Windows et UNIX System Services (USS) sous z/OS. Exécutez cette commande à l'invite du système d'exploitation ou dans un script de shell.

Pour plus d'informations sur l'utilisation de la commande MVS MODIFY pour envoyer des commandes à un programme Q Capture en cours d'exécution sous z/OS, voir Gestion des programmes de réplication Q et de publication d'événements en cours d'exécution à l'aide de la commande MVS MODIFY.

Syntaxe



Paramètres :



Paramètres

Le tableau 40, à la page 408 définit les paramètres d'appel pour la commande **asnlccmd**.

Tableau 40. Définitions des paramètres d'appel *asnlccmd*

Paramètre	Définition
<code>capture_server=nom_bd</code>	<p>Définit le nom de la base de données ou du sous-système contenant les tables de contrôles Q Capture.</p> <p>z/OS Spécifie le nom du sous-système DB2 sur lequel le programme Q Capture s'exécutera. Pour le partage de données, il se peut que la valeur de capture_server corresponde au nom de liaison de groupe utilisé pour exécuter Q Capture dans n'importe quel LPAR sans modifier le langage JCL de la tâche démarrée.</p> <p>Linux UNIX Windows Si vous n'indiquez pas de serveur de contrôle de Capture, ce paramètre prend par défaut la valeur de la variable d'environnement <i>DB2DBDFT</i>.</p>
<code>capture_schema=schéma</code>	Définit un nom identifiant le programme Q Capture avec lequel vous voulez travailler.
<code>chgparms</code>	<p>Indique de modifier un ou plusieurs des paramètres d'exploitation suivants d'un programme Q Capture pendant son exécution :</p> <ul style="list-style-type: none"> • autostop • commit_interval • logreuse • logstdout • memory_limit • monitor_interval • monitor_limit • prune_interval • qfull_num_retries • qfull_retry_delay • signal_limit • sleep_interval • term • trace_limit <p>Important : Le paramètre modifié doit suivre immédiatement le paramètre chgparms.</p> <p>z/OS Restriction : Il est impossible de modifier la valeur du paramètre memory_limit lorsque le programme Q Capture est en cours d'exécution. Pour modifier cette valeur, vous devez arrêter le programme Q Capture.</p> <p>Vous pouvez définir plusieurs paramètres dans une commande asnlccmd chgparms et vous pouvez modifier les valeurs de ces paramètres aussi souvent que vous le souhaitez. Les modifications substituent provisoirement les valeurs dans la table <i>IBMQREP_CAPPARMS</i>, mais ne les écrivent pas dans la table. Lorsque vous arrêtez puis redémarrez le programme Q Capture, ce dernier utilise les valeurs affectées à <i>IBMQREP_CAPPARMS</i>. La rubrique «Descriptions des paramètres <i>asnlccap</i>», à la page 375 comprend des détails sur les paramètres que vous pouvez remplacer à l'aide de cette commande.</p>

Tableau 40. Définitions des paramètres d'appel *asncqcmd* (suite)

Paramètre	Définition
prune	Indique d'ordonner à un programme Q Capture d'élaguer les tables IBMQREP_CAPMON, IBMQREP_CAPQMON, IBMQREP_CAPTRACE et IBMQREP_SIGNAL une fois. Cet élagage s'ajoute aux élagages réguliers planifiés définis par le paramètre prune_interval .
qryparms	Indique si vous souhaitez les valeurs actuelles des paramètres d'exploitation Q Capture écrits sur la sortie standard (stdout).
reinit	Indique que le programme Q Capture désactive puis active tous les abonnements Q et les publications à l'aide des dernières valeurs dans les tables IBMQREP_SUBS, IBMQREP_SRC_COLS et IBMQREP_SENDQUEUEUES. Cette commande vous permet de modifier les attributs de plusieurs abonnements Q ou publications pendant que le programme Q Capture est en cours d'exécution. Cette commande ne demande pas un nouveau chargement des cibles.
reinitq=file_envoi	Indique que le programme Q Capture actualise une file d'attente d'envoi à l'aide des derniers attributs de la table IBMQREP_SENDQUEUEUES. Cette commande affecte tous les abonnements Q ou toutes les publications qui utilisent cette file d'attente d'envoi. Seuls les attributs suivants seront actualisés : ERROR_ACTION, HEARTBEAT_INTERVAL, MAX_MESSAGE_SIZE.
startq=file_attente_envoi/all	Indique de commencer à placer des messages dans l'une ou la totalité des files d'attente d'envoi désactivées. Q Capture définit l'état de la file d'attente sur Actif (A) et recommence à placer des messages dans une file d'attente définie ou dans toutes les files d'attente inactives. Q Capture relance la lecture du journal au point de redémarrage le plus ancien parmi toutes les files d'attente d'envoi et reprend les files d'attente arrêtées jusqu'à ce que toutes les files d'attentes aient le même point de redémarrage.
status	A indiquer pour recevoir les messages donnant l'état de chaque unité d'exécution Q Capture (main, administration, prune, holdl, transaction, et worker).

Tableau 40. Définitions des paramètres d'appel *asncqcmd* (suite)

Paramètre	Définition
show details	<p>Indique après le paramètre status d'afficher un rapport plus détaillé de l'état du programme Q Capture comportant les informations suivantes :</p> <ul style="list-style-type: none"> • Si le programme Q Capture est en cours d'exécution • Temps écoulé depuis le démarrage du programme • Emplacement du journal de diagnostic Q Capture • Nombre d'abonnements Q actifs • Valeurs de CURRENT_LOG_TIME et CURRENT_MEMORY. Ces valeurs peuvent être plus récentes que ce qui est inséré dans la table de contrôle IBMQREP_CAPMON. • LSN logique de la dernière transaction que le programme Q Capture a publié vers une file d'attente d'envoi • Quantité de mémoire en mégaoctets que Q Capture a utilisée pendant le dernier intervalle de surveillance pour générer les transactions à partir des enregistrements de journal <p style="text-align: center;">Linux UNIX Windows</p> <ul style="list-style-type: none"> • Chemin d'accès aux fichiers journaux DB2 • Fichier journal DB2 le plus ancien nécessaire pour un redémarrage Q Capture • Fichier journal DB2 en cours capturé
stop	<p>Indique d'arrêter le programme Q Capture dans l'ordre et de valider les messages qu'il a traité jusque là.</p> <p>Vous pouvez spécifier le paramètre captureupto, le paramètre stopafter, voire les deux, afin d'arrêter Q Capture d'une façon contrôlée à partir d'un point indiqué :</p> <p>captureupto Indiquez une valeur d'horodatage complète ou partielle pour ordonner à Q Capture d'arrêter la lecture du journal à un point particulier, puis de s'arrêter. Vous pouvez également indiquer CURRENT_TIMESTAMP ou EOL pour que Q Capture s'arrête après la fin du journal actif.</p> <p>stopafter Indique que Q Capture s'arrête lorsque l'une des conditions suivantes est vraie :</p> <p>data_applied Q Capture s'arrête après que toutes les modifications situées après le point d'arrêt spécifié sont appliquées à la cible.</p> <p>data_sent Q Capture s'arrête après la suppression de tous les messages de la file d'attente de transmission pointant vers le système cible ou après le traitement de tous les messages par le programme Q Apply dans le cas d'une file d'attente de réception/d'envoi locale partagée.</p>

Tableau 40. Définitions des paramètres d'appel *asnqcmd* (suite)

Paramètre	Définition
stopq=file_attente_envoi/all	<p>Indique d'arrêter l'insertion des messages dans l'une ou la totalité des files d'attente d'envoi. définit l'état de la file d'attente sur Inactive (I) et arrête l'insertion des messages dans une file d'attente définie ou dans toutes les files d'attente. Le programme Q Capture continue à publier des modifications pour les abonnements Q associés à des files d'attente d'envoi actives. Si toutes les files d'attente d'envoi sont arrêtées, Q Capture continue à lire le journal pour rechercher des signaux tels que CAPSTART, continue à effectuer des insertions dans ses tables de contrôle et attend les commandes.</p> <p>Outre une file d'attente individuelle, vous pouvez spécifier le paramètre captureupto parameter, le paramètre stopafter, voire les deux, afin que Q Capture arrête de placer des messages en file d'attente d'une façon contrôlée à partir d'un point indiqué :</p> <p>captureupto Indiquez une valeur d'horodatage complète ou partielle pour ordonner à Q Capture de publier jusqu'à un point particulier, puis d'arrêter de placer des messages en file d'attente. Vous pouvez également indiquer CURRENT_TIMESTAMP ou EOL pour que Q Capture arrête de placer des messages en file d'attente après la fin du journal actif.</p> <p>stopafter Indique que Q Capture arrête de placer des messages en file d'attente lorsque l'une des conditions suivantes est vraie :</p> <p>data_applied Toutes les modifications situées après le point d'arrêt spécifié sont appliquées à la cible.</p> <p>data_sent Tous les messages sont supprimées de la file d'attente de transmission pointant vers le système cible ou le programme Q Apply a traité tous les messages dans le cas d'une file d'attente de réception/d'envoi locale partagée.</p>

Exemple 1

Pour donner l'instruction au programme Q Capture d'actualiser tous les abonnements Q et toutes les publications à l'aide des dernières valeurs dans les tables de contrôle Q Capture :

```
asnqcmd capture_server=sourcedb capture_schema="alpha" reinit
```

Cette commande démarrera toutes les files d'attente inactives si le programme Q Capture a été démarré avec **startallq=0** (paramètre par défaut).

Exemple 2

Pour donner l'instruction au programme Q Capture d'actualiser les attributs `ERROR_ACTION`, `HEARTBEAT_INTERVAL` et `MAX_MESSAGE_SIZE` de tous les abonnements Q et publications qui utilisent une file d'attente d'envoi appelée Q1 :

```
asnqccmd capture_server=sourcedb capture_schema="alpha" reinitq="Q1"
```

Exemple 3

Pour raccourcir provisoirement à une minute l'intervalle d'élagage par défaut d'un programme Q Capture en cours d'exécution et augmenter provisoirement la durée de veille par défaut du programme Q Capture après le traitement des abonnements Q et des publications :

```
asnqccmd capture_server=sourcedb capture_schema="alpha" chgparms  
prune_interval=60 sleep_interval=10000
```

Exemple 4

Pour recevoir des messages plus détaillés sur l'état du programme Q Capture :

```
asnqccmd capture_server=sourcedb capture_schema="alpha" status show details
```

Exemple 5

Pour que le programme Q Capture arrête de lire le journal de récupération au 15 octobre 2010 à 15 h 30, attendez que toutes les transactions validées dans la base de données cible jusqu'à cette date et heure précises soient appliquées, puis arrêtez le traitement :

```
asnqccmd capture_server=sourcedb capture_schema="alpha" stop  
captureupto="2010-10-15-15.30" stopafter=data_applied
```

Exemple 6

Pour ordonner au programme Q Capture d'arrêter la lecture du journal de récupération immédiatement puis de s'arrêter après la suppression de tous les messages de la file d'attente de transmission (ou de la file d'attente d'envoi/de réception locale partagée) :

```
asnqccmd capture_server=sourcedb capture_schema="alpha" stop stopafter=data_sent
```

asnqccmd : Gestion d'un programme Q Capture en cours d'exécution sur des bases de données Oracle

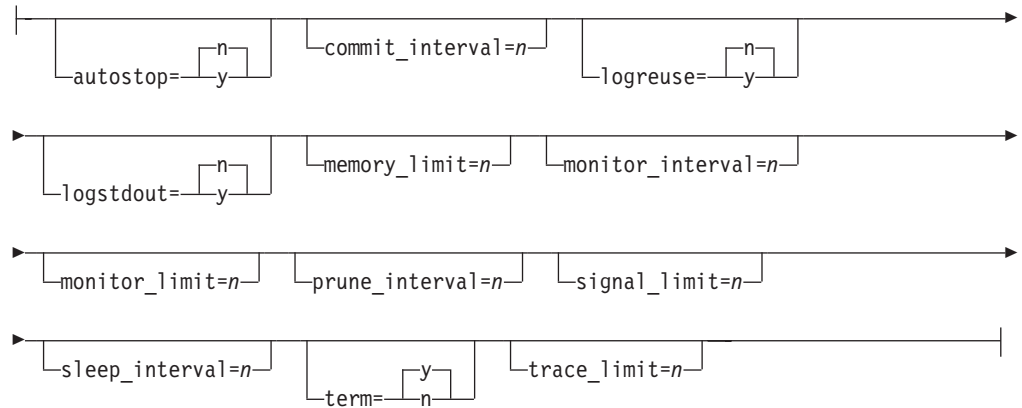
Utilisez la commande **asnqccmd** pour envoyer une commande à un programme Q Capture qui s'exécute sur une base de données Oracle sur un système Linux, UNIX, ou Windows. Exécutez cette commande à l'invite du système d'exploitation ou dans un script de shell.

Syntaxe

```
▶▶ asnqccmd—capture_server=nom_bd—capture_schema=schéma—▶▶
```




Paramètres :



Paramètres

Le tableau suivant définit les paramètres d'appel pour la commande **asnoqcmd**.

Tableau 41. Définitions pour les paramètres d'appel **asnoqcmd**

Paramètre	Définition
capture_server = <i>nom_bd</i>	Indique le nom de la base de données Oracle qui contient les tables de contrôle Q Capture. Si vous n'indiquez pas de serveur Q Capture, ce paramètre aura comme valeur par défaut la valeur de la variable d'environnement <i>ORACLE_SID</i> .
capture_schema = <i>schéma</i>	Définit un nom identifiant le programme Q Capture avec lequel vous voulez travailler. Le schéma par défaut est ASN.

Tableau 41. Définitions pour les paramètres d'appel *asnoqcmd* (suite)

Paramètre	Définition
chgparms	<p>Spécifiez ce paramètre pour modifier un ou plusieurs des paramètres opérationnels suivants d'un programme Q Capture pendant qu'il est en cours d'exécution sur une base de données Oracle :</p> <ul style="list-style-type: none"> • autostop • commit_interval • logreuse • logstdout • memory_limit • monitor_interval • monitor_limit • prune_interval • qfull_num_retries • qfull_retry_delay • signal_limit • sleep_interval • trace_limit <p>Vous pouvez spécifier plusieurs paramètres dans une commande asnoqcmd chgparms, et vous pouvez modifier ces valeurs de paramètre aussi souvent que vous le souhaitez. Les modifications substituent provisoirement les valeurs dans la table IBMQREP_CAPPARMS, mais ne les écrivent pas dans la table. Lorsque vous arrêtez puis redémarrez le programme Q Capture, ce dernier utilise les valeurs affectées à IBMQREP_CAPPARMS. «asnoqcap : Démarrage d'un programme Q Capture pour une base de données Oracle», à la page 398 comprend des descriptions des paramètres que vous pouvez écraser avec cette commande.</p> <p>Condition require : Le paramètre modifié doit suivre immédiatement le paramètre chgparms.</p>
prune	<p>Spécifiez ce paramètre pour demander au programme Q Capture d'élaguer une fois les tables IBMQREP_CAPMON, IBMQREP_CAPQMON, IBMQREP_CAPTRACE, et IBMQREP_SIGNAL. Cet élagage s'ajoute aux élagages réguliers planifiés définis par le paramètre prune_interval.</p>
qryparms	<p>Spécifiez ce paramètre si vous voulez que les valeurs actuelles du paramètre opérationnel d'un programme Q Capture soient écrites sur la sortie standard (stdout).</p>
reinit	<p>Spécifiez ce paramètre pour que le programme Q Capture désactive puis active tous les abonnements Q et publications à l'aide des dernières valeurs des tables IBMQREP_SUBS, IBMQREP_SRC_COLS, et IBMQREP_SENDQUEUES. Cette commande vous permet de modifier les attributs de plusieurs abonnements Q ou publications pendant que le programme Q Capture est en cours d'exécution. Cette commande ne demande pas un nouveau chargement des cibles.</p>

Tableau 41. Définitions pour les paramètres d'appel `asnoqcmd` (suite)

Paramètre	Définition
<code>reinitq=file_envoi</code>	Spécifiez ce paramètre pour que le programme Q Capture régénère une file d'attente d'envoi à l'aide des derniers attributs de la table <code>IBMQREP_SENDQUEUEES</code> . Cette commande affecte tous les abonnements Q ou toutes les publications qui utilisent cette file d'attente d'envoi. Seuls les attributs suivants seront actualisés : <code>ERROR_ACTION</code> , <code>HEARTBEAT_INTERVAL</code> , <code>MAX_MESSAGE_SIZE</code> .
<code>status</code>	Spécifiez ce paramètre pour recevoir des messages indiquant l'état de chaque unité d'exécution Q Capture (<code>main</code> , <code>administration</code> , <code>prune</code> , <code>holdl</code> , <code>worker</code> , et <code>transaction</code>).
<code>show details</code>	Spécifiez ce paramètre après le paramètre <code>status</code> pour afficher un rapport plus détaillé concernant le statut du programme Q Capture, avec les informations suivantes : <ul style="list-style-type: none"> • Si le programme Q Capture est en cours d'exécution • Temps écoulé depuis le démarrage du programme • Emplacement du journal de diagnostic Q Capture • Nombre d'abonnements Q actifs • Valeurs de <code>CURRENT_LOG_TIME</code> et <code>CURRENT_MEMORY</code>. Ces valeurs peuvent être plus récentes que celles insérées dans la table de contrôle <code>IBMQREP_CAPMON</code>. • LSN logique de la dernière transaction que le programme Q Capture a publiée vers une file d'attente d'envoi. Ce numéro s'affiche en tant que translation hexadécimale du numéro décimal de modification de système (SCN) Oracle d'origine. • Quantité de mémoire en mégaoctets que Q Capture a utilisée pendant le dernier intervalle de surveillance pour générer les transactions à partir des enregistrements de journal • Chemin d'accès vers les fichiers journaux redo Oracle • Fichier journal redo Oracle le plus ancien qui est nécessaire pour le redémarrage de Q Capture • Fichier journal redo Oracle en cours capturé
<code>stop</code>	Spécifiez ce paramètre pour arrêter le programme Q Capture dans l'ordre et valider les messages qu'il a traités jusque-là.

Exemple - Régénérer tous les abonnements Q et publications

Pour donner l'instruction au programme Q Capture de régénérer tous les abonnements Q et toutes les publications à l'aide des dernières valeurs dans les tables de contrôle Q Capture, émettez la commande suivante :

```
asnoqcmd capture_server=sourcedb capture_schema="alpha" reinit
```

Exemple - Régénérer les attributs de file d'attente d'envoi

Pour donner l'instruction au programme Q Capture de régénérer les attributs `ERROR_ACTION`, `HEARTBEAT_INTERVAL` et `MAX_MESSAGE_SIZE` de tous les abonnements Q et publications qui utilisent une file d'attente d'envoi appelée `Q1`, émettez la commande suivante :

```
asnoqcmd capture_server=sourcedb capture_schema="alpha" reinitq="Q1"
```

Exemple - Modifier les paramètres spécifiés

Pour raccourcir provisoirement à une minute l'intervalle d'élagage par défaut d'un programme Q Capture en cours d'exécution et augmenter provisoirement la durée de veille par défaut du programme Q Capture après le traitement des abonnements Q et des publications, émettez la commande suivante :

```
asnoqcmd capture_server=sourcedb capture_schema="alpha" chgparms  
prune_interval=60 sleep_interval=10000
```

Exemple - Afficher les détails du statut Q Capture

Pour recevoir des messages détaillés sur l'état du programme Q Capture, émettez la commande suivante :

```
asnoqcmd capture_server=sourcedb capture_schema="alpha" status show details
```

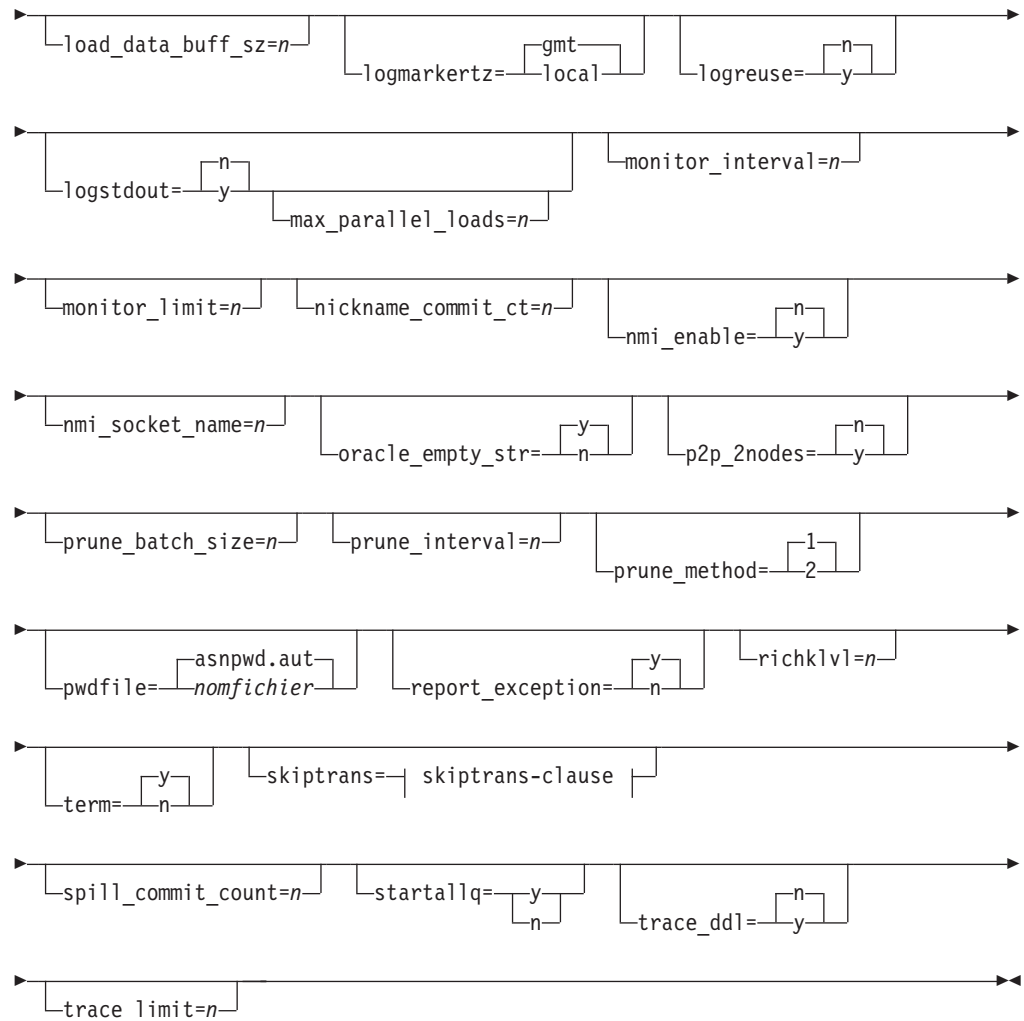
asnqapp : lancement d'un programme Q Apply

Pour lancer un programme Q Apply, utilisez la commande **asnqapp** avec les services système Linux, UNIX, Windows et UNIX sous z/OS. Exécutez cette commande à l'invite du système d'exploitation ou dans un script de shell. Les paramètres de démarrage que vous définissez ne s'appliquent qu'à cette session.

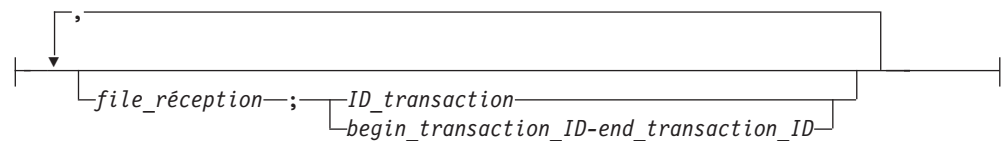
Une fois que vous avez lancé le programme Q Apply, il s'exécute en continu jusqu'à ce que vous l'arrêtiez ou qu'il détecte une erreur irrémédiable.

Syntaxe





skiptrans-clause ::



Descriptions des paramètres asnqapp

Ces descriptions fournissent des détails sur les paramètres asnqapp, leurs valeurs par défaut et les raisons susceptibles de vous obliger à modifier la valeur par défaut de votre environnement.

- «apply_server»
- «apply_schema»
- «apply_path», à la page 419
- «applydelay», à la page 420
- «applyupto», à la page 420
- «arm», à la page 422
- «autostop», à la page 422
- «buffered_inserts», à la page 422
- «caf», à la page 423
- «commit_count», à la page 423
- «deadlock_retries», à la page 424
- «dftmodelq», à la page 424
- «ignbaddata», à la page 425
- «insert_bidi_signal», à la page 425
- «loadcopy_path», à la page 426
- «load_data_buff_sz», à la page 426
- «logmarkertz», à la page 426
- «logreuse», à la page 427
- «logstdout», à la page 428
- «max_parallel_loads», à la page 428
- «monitor_interval», à la page 428
- «monitor_limit», à la page 429
- «nickname_commit_ct», à la page 430
- «p2p_2nodes», à la page 430
- «pwdfile», à la page 432
- «prune_interval», à la page 431
- «report_exception», à la page 432
- «richklvl», à la page 432
- «skiptrans», à la page 433
- «spill_commit_count», à la page 433
- «startallq», à la page 434
- «term», à la page 434
- «trace_limit», à la page 435
- «trace_ddl», à la page 434

apply_server

z/OS

Valeur par défaut : aucune

Linux UNIX Windows

Valeur par défaut : **apply_server**=valeur de la variable d'environnement DB2DBDFT, si elle est définie

Le paramètre **apply_server** identifie la base de données ou le sous-système où un programme Q Apply est exécuté et où ses tables de contrôle sont stockées. Les tables de contrôle contiennent des informations sur les cibles, les abonnements Q, les files d'attente WebSphere MQ et les préférences utilisateur. Le serveur Q Apply doit correspondre à la base de données ou au sous-système contenant les cibles.

apply_schema

Valeur par défaut : **apply_schema**=ASN

Le paramètre **apply_schema** vous permet de faire la distinction entre plusieurs instances du programme Q Apply sur un serveur Q Apply.

Le schéma identifie un programme Q Apply et ses tables de contrôle. Deux programmes Q Apply dotés du même schéma ne peuvent pas être exécutés sur un serveur.

Un seul programme Q Apply peut créer plusieurs unités d'exécution du navigateur. Chaque unité d'exécution du navigateur lit les messages d'une seule file d'attente de réception. Pour cette raison, il n'est pas nécessaire de créer plusieurs instances du programme Q Apply sur un serveur pour répartir le flux de données actuellement appliqué aux cibles.

z/OS

Sous z/OS, aucun caractère spécial n'est autorisé dans le schéma Q Apply, à l'exception du trait de soulignement (_).

apply_path

Valeur par défaut : aucune

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **apply_path** indique le répertoire dans lequel un programme Q Apply stocke ses fichiers de travail et son fichier journal. Par défaut, le chemin est le répertoire dans lequel vous démarrez le programme. Vous pouvez modifier ce chemin.

Windows

Si vous démarrez un programme Q Apply en tant que service Windows, ce programme démarre par défaut depuis le répertoire \SQLLIB\bin.

z/OS

Le programme Q Apply étant une application POSIX, le chemin par défaut dépend du mode de démarrage du programme :

- Si vous démarrez un programme Q Apply à partir d'une invite de ligne de commande USS, le chemin est le répertoire de démarrage du programme.
- Si vous démarrez un programme Q Apply à l'aide d'une tâche démarrée ou via JCL, le chemin par défaut est le répertoire de base dans le système de fichiers USS de l'ID utilisateur associé à la tâche ou au travail démarré.

Pour changer de chemin, vous pouvez spécifier un nom de chemin ou un qualificatif HLQ, tel que //QAPP. Lorsque vous utilisez un HLQ, des fichiers séquentiels sont créés conformément aux conventions de dénomination z/OS pour ce type de fichiers. Les fichiers séquentiels dépendent de l'ID utilisateur qui exécute le programme. Sinon, ces noms de fichiers sont identiques à ceux qui sont stockés dans un chemin de répertoire explicitement nommé (le qualificatif HLQ est dans ce cas concaténé en première partie du nom de fichier). Par exemple, sysadm.QAPPV9.filename. L'utilisation d'un qualificatif HLQ peut être utile si vous voulez que les fichiers LOADMSG et les fichiers journaux Q Apply soient gérés par le système (SMS).

Si vous voulez que la tâche lancée par Q Apply écrive sur un ensemble de données .log avec un ID utilisateur différent de l'ID qui exécute la tâche (par exemple TSOUSER), vous devez indiquer une marque de guillemet simple (') comme caractère d'échappement lorsque vous utilisez le format SYSIN pour les paramètres d'entrée sur la tâche démarrée. Par exemple, si vous vouliez utiliser le qualificatif de haut niveau JOESMITH, alors l'ID utilisateur TSOUSER qui exécute le programme Q Apply doit bénéficier des droits RACF pour écrire des fichiers à l'aide du qualificatif de haut niveau JOESMITH, comme dans l'exemple suivant :

```
//SYSIN DD *  
  APPLY_PATH=/'JOESMITH  
/*
```

Vous pouvez définir le paramètre **apply_path** lors du démarrage du programme Q Apply ou modifier sa valeur enregistrée dans la table IBMQREP_APPLYPARMS. Vous ne pouvez pas modifier ce paramètre pendant l'exécution du programme Q Apply.

applydelay

Valeur par défaut : `applydelay=0` (seconde)

Méthode de modification : au démarrage de Q Apply

Le paramètre **applydelay** contrôle la durée d'attente (exprimée en secondes) du programme Q Apply avant la réexécution de chaque transaction sur la cible. Le délai est basé sur l'heure de validation de la transaction source. Q Apply retarde l'application des transactions jusqu'à ce que l'heure en cours atteigne ou dépasse l'heure de validation de la transaction source à laquelle est ajoutée la valeur du paramètre **applydelay**. Les modifications effectuées dans la base de données source sont capturées et envoyées dans la file d'attente de réception, où elles attendent pendant la période ainsi définie.

Ce paramètre peut être utilisé, par exemple, pour conserver plusieurs copies de la base de données source à des dates et heures différentes pour une reprise en cas de problème sur le système source. Par exemple, si un utilisateur supprime accidentellement des données sur le système principal, les données restent disponibles car il existe une copie de la base de données.

Le paramètre **applydelay** n'a aucun effet sur les paramètres **applyupto** ou **autostop**.

Important : Si vous prévoyez d'utiliser le paramètre **applydelay**, assurez-vous que la file d'attente de réception possède suffisamment d'espace pour conserver les messages qui s'accumulent pendant la période de retard.

applyupto

Valeur par défaut : aucune

Méthode de modification : au démarrage de Q Apply

Le paramètre **applyupto** identifie un horodatage qui donne l'instruction au programme Q Apply de s'arrêter après le traitement des transactions qui ont été validées à la source aux heures spécifiées ou avant :

- Un horodatage spécifique que vous indiquez
- Le mot clé `CURRENT_TIMESTAMP`, qui désigne l'heure à laquelle le programme Q Apply a commencé

Vous pouvez en option spécifier les mots clés `WAIT` ou `NOWAIT` pour contrôler l'arrêt de Q Apply :

WAIT (default)

Q Apply ne s'arrête pas tant qu'il ne reçoit pas et n'a pas traité toutes les transactions jusqu'à l'horodatage GMT spécifié ou la valeur de `CURRENT_TIMESTAMP`, même si la file d'attente de réception devient vide.

NOWAIT

Q Apply s'arrête après avoir traité toutes les transactions de la file d'attente de réception, même s'il n'a pas vu une transaction avec un horodatage de validation qui correspond ou dépasse l'horodatage GMT spécifié ou la valeur de `CURRENT_TIMESTAMP`.

Le paramètre **applyupto** s'applique à toutes les unités d'exécution du navigateur d'une instance Q Apply. Chaque unité d'exécution du navigateur s'arrête lorsqu'elle lit un message sur sa file d'attente de réception avec un horodatage de validation source qui correspond au temps spécifié ou le dépasse. Le programme Q Apply s'arrête lorsque toutes ses unités d'exécution du navigateur déterminent que toutes les transactions avec un horodatage de validation source antérieures à l'horodatage **applyupto** ou incluant cet horodatage ont été appliquées. Toutes les transactions ayant un horodatage de validation source supérieur à l'horodatage GMT spécifié restent sur la file d'attente de réception et seront traitées lors de la prochaine exécution du programme Q Apply.

Cet horodatage doit être spécifié en tant que format d'horodatage partiel ou complet en temps GMT (Greenwich Mean Time). L'horodatage complet utilise le format suivant : YYYY-MM-DD-HH.MM.SS.mmmmm. Par exemple 2007-04-10-10.35.30.555555 correspond à l'horodatage GMT du 10 avril 2007, à 10 heures, 35 minutes, 30 secondes et 555555 microsecondes.

Vous pouvez spécifier l'horodatage partiel en utilisant l'un des formats suivants :

YYYY-MM-DD-HH.MM.SS

Par exemple, 2007-04-10-23.35.30 correspond à l'horodatage GMT partiel du 10 avril 2007, à 23 heures, 35 minutes et 30 secondes.

YYYY-MM-DD-HH.MM

Par exemple, 2007-04-10-14.30 correspond à l'horodatage GMT partiel du 10 avril 2007, à 14 heures et 30 minutes.

YYYY-MM-DD-HH

Par exemple, 2007-04-10-01 correspond à l'horodatage GMT partiel du 10 avril 2007, à 01 heure du matin.

HH.MM Par exemple, 14:55 correspond à l'horodatage GMT partiel de la journée en cours à 14 heures et 55 minutes.

HH Par exemple, 14 correspond à l'horodatage GMT partiel de la journée en cours à 14 heures.

L'horodatage partiel pourrait être utilisé pour spécifier une heure au format HH.MM. Ce format peut s'avérer utile si vous planifiez une tâche pour démarrer le programme Q Apply chaque jour à 1 heure du matin PST (Pacific Standard Time) et que vous voulez arrêter le programme après avoir traité les transactions ayant été validées à la source avec un horodatage GMT sur l'horodatage PST à 4 heures du matin ou avant. Par exemple, exécutez la tâche suivante à 1 heure du matin PST et définissez le paramètre **applyupto** pour terminer la tâche à 4 heures du matin PST :

```
asnqapp apply_server=MYTESTSERVER apply_schema=ASN applyupto=12.00
```

Lors de l'heure d'été, la différence entre le temps GMT et l'heure locale varie en fonction de votre zone géographique. Par exemple, le fuseau horaire pacifique correspond à GMT-8 heures en automne et en hiver. Lors de l'heure d'été, au printemps et en été, il correspond à GMT-7 heures.

Restriction : Vous ne pouvez pas spécifier à la fois le paramètre **autostop** et le paramètre **applyupto**.

Il se peut que vous souhaitiez définir l'intervalle des pulsations sur une valeur supérieure à zéro de telle manière que le programme Q Apply puisse signaler si l'heure indiquée dans le paramètre **applyupto** est dépassée.

arm

Valeur par défaut : aucune

Méthode de modification : au démarrage de Q Apply

Vous pouvez utiliser le paramètre **arm=identificateur** pour définir un identificateur unique pour le programme Q Apply que le gestionnaire de redémarrage automatique utilise pour démarrer automatiquement une instance Q Apply arrêtée. La valeur alphanumérique à trois caractères que vous renseignez est associée au nom d'élément ARM que le programme Q Apply génère pour lui-même : `ASNQAxxxxyyy` (où `xxxx` est le nom de liaison du groupe de partage de données et `yyy` est le nom du membre DB2). Vous pouvez spécifier toute longueur de chaîne pour le paramètre **arm**, mais le programme Apply concatènera uniquement trois caractères maximum au nom en cours. Si nécessaire, le programme Q Apply remplira le nom par des blancs pour composer un nom unique de 16 octets.

autostop

Valeur par défaut : **autostop=n**

Méthodes de modification : au démarrage de Q Apply ; lors de l'exécution de Q Apply ; table `IBMQREP_APPLYPARMS`

Le paramètre **autostop** vous permet de demander à un programme Q Apply de s'arrêter automatiquement quand aucune transaction ne doit être appliquée. Par défaut (**autostop=n**), un programme Q Apply continue de s'exécuter lorsque les files d'attente sont vides et attend l'arrivée de transactions.

En règle générale, le programme Q Apply est exécuté en permanence chaque fois que la base de données cible est active. Par conséquent, dans la plupart des cas, vous devez conserver la valeur par défaut (**autostop=n**). Définissez **autostop=y** uniquement dans les cas où le programme Q Apply est exécuté à intervalles réguliers, par exemple, lorsque vous synchronisez des systèmes connectés de manière irrégulière ou dans des scénarios de test.

Si vous définissez le paramètre **autostop=y**, le programme Q Apply s'arrête dès que toutes les files d'attente de réception ont été vidées une seule fois. Lorsque l'unité d'exécution du navigateur de chaque file d'attente de réception détecte l'absence de messages dans la file d'attente, l'unité d'exécution cesse la lecture de la file d'attente. Le programme Q Apply s'arrête une fois toutes les unités d'exécution arrêtées. Les messages peuvent continuer d'arriver dans les files d'attente pour lesquelles l'unité d'exécution du navigateur s'est arrêtée, mais ils seront collectés jusqu'au redémarrage du programme Q Apply.

Restriction : Vous ne pouvez pas spécifier à la fois le paramètre **autostop** et le paramètre **applyupto**.

buffered_inserts

Valeur par défaut : **buffered_inserts=n**

Méthode de modification : au démarrage de Q Apply

Le paramètre **buffered_inserts** indique si le programme Q Apply utilise des insertions en mémoire tampon, ce qui permet d'améliorer les performances dans certaines bases de données partitionnées avec de nombreuses opérations INSERT. Si vous spécifiez **buffered_inserts=0**, Q Apply relie en interne les fichiers appropriés avec l'option INSERT BUF. Cette option de liaison permet au noeud coordinateur d'une base de données partitionnée d'accumuler les lignes insérées dans des mémoires tampon au lieu de les transférer immédiatement à leurs partitions cibles. Lorsqu'une mémoire tampon est saturée, ou lorsqu'une instruction SQL telle que UPDATE, DELETE ou INSERT vers une autre table ou une instruction COMMIT/ROLLBACK est rencontrée, toutes les lignes de la mémoire tampon sont envoyées ensemble dans la partition cible.

Vous pouvez améliorer encore les performances en combinant l'utilisation d'insertions en mémoire tampon et le paramètre **commit_count**.

Si les insertions en mémoire tampon sont activées, le programme Q Apply n'assure pas le traitement des exceptions. En cas de conflit ou d'erreur, le programme Q Apply arrête de lire dans la file d'attente. Pour recommencer la lecture après l'exception, vous devez lancer le traitement des messages de la file d'attente et démarrer le programme Q Apply avec le paramètre **buffered_inserts=0**.

z/OS

caf

Valeur par défaut : **caf=n**

Méthode de modification : au démarrage de Q Apply

Le programme Q Apply s'exécute avec une connexion RRS. Vous pouvez remplacer cette valeur par défaut et indiquer au programme Q Apply d'utiliser la fonction de connexion d'appel CAF en définissant l'option **caf =y**.

Si RRS n'est pas disponible, un message vous l'indique et le programme Q Apply utilise la connexion CAF. Le message vous avertit que le programme n'est pas parvenu à initialiser une connexion car RRS n'est pas démarré. Le programme tente d'utiliser une connexion CAF à la place. Le programme s'exécute correctement avec la connexion CAF.

commit_count

Valeur par défaut : **commit_count=1**

Méthode de modification : au démarrage de Q Apply

Le paramètre **commit_count** définit le nombre de transactions que chaque unité d'exécution agent Q Apply applique à la table cible dans une portée de validation. Par défaut, l'unité d'exécution agent valide après chaque transaction qu'il applique.

En augmentant la valeur du paramètre **commit_count** et en regroupant davantage de transactions dans la portée de validation, vous pouvez améliorer les performances.

Recommandation : N'utilisez une valeur plus élevée pour **commit_count** que dans le cas d'un verrouillage au niveau de la ligne. Définissez ce paramètre avec soin

lorsqu'il est utilisé avec un grand nombre d'unités d'exécution agent, il peut générer une escalade de verrous et provoquer des dépassements du délai d'attente des verrous et des blocages de relances.

deadlock_retries

Valeur par défaut : `deadlock_retries=3`

Méthodes de modification : au démarrage de Q Apply ; lors de l'exécution de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **deadlock_retries** indique le nombre de fois que le programme Q Apply essaie de réappliquer les changements dans les tables cible dans le cas d'un blocage SQL ou d'un dépassement du délai de verrouillage. La valeur par défaut est trois tentatives. Ce paramètre contrôle également le nombre de fois que le programme Q Apply essaie d'insérer, de mettre à jour ou de supprimer des lignes de ses tables de contrôle après un blocage SQL.

Une fois la limite atteinte, si les blocages persistent, l'unité d'exécution du navigateur s'arrête. Il sera peut-être nécessaire de définir une valeur supérieure pour le paramètre **deadlock_retries** si des applications mettent fréquemment à jour la base de données cible et en présence d'un haut niveau de conflit. Ou, si vous disposez d'un nombre important de files d'attente de réception et d'unités d'exécution du navigateur correspondantes, la définition d'une valeur supérieure pour le paramètre **deadlock_retries** permet de résoudre des conflits éventuels dans les environnements d'homologues et autres environnements de réplication multidirectionnelle, ainsi que dans des tables de contrôle, telles que la table IBMQREP_DONEMSG.

Restriction : La valeur par défaut du paramètre **deadlock_retries** ne peut pas être diminuée.

dftmodelq

Valeur par défaut : aucune

Méthode de modification : au démarrage de Q Apply

Par défaut, le programme Q Apply utilise IBMQREP.SPILL.MODELQ comme nom pour la file d'attente modèle qu'il utilise pour créer des files d'attente auxiliaires pour le processus de chargement. Pour définir un autre nom de file d'attente modèle par défaut, définissez le paramètre **dftmodelq**. La liste suivante récapitule le comportement du paramètre :

Si vous spécifiez dftmodelq lorsque vous démarrez Q Apply

Pour chaque abonnement Q, Q Apply vérifiera si vous avez spécifié un nom de file d'attente modèle pour l'abonnement Q en consultant la valeur de la colonne MODELQ dans la table de contrôle IBMQREP_TARGETS :

- Si la valeur est NULL ou IBMQREP.SPILL.MODELQ, alors Q Apply utilisera la valeur que vous spécifiez pour le paramètre **dftmodelq**.
- Si la colonne contient une autre valeur non NULL, alors Q Apply utilisera la valeur de la colonne MODELQ et ignorera la valeur que vous spécifiez pour le paramètre **dftmodelq**.

Si vous ne spécifiez pas `dftmodelq` lorsque vous démarrez Q Apply
Q Apply utilisera la valeur de la colonne MODELQ dans la table IBMQREP_TARGETS. Si la valeur est NULL, Q Apply utilisera par défaut IBMQREP.SPILL.MODELQ.

ignbaddata

Valeur par défaut : aucune

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Remarque : Ce paramètre s'applique uniquement si le programme Q Apply utilise les Composants internationaux pour l'Unicode (ICU) pour la conversion de page de code (si la page de code de la base de données source et la page de code utilisée par Q Apply sont différentes).

Le paramètre **ignbaddata** indique si le programme Q Apply vérifie les caractères non autorisés dans les données de la source et continue le traitement si c'est le cas.

Si vous indiquez **ignbaddata=y**, Q Apply vérifie l'existence de caractères non autorisés et prend les mesures suivantes s'il en détecte :

- N'applique pas la ligne avec les caractères non autorisés.
- Insère une ligne dans la table IBMQREP_EXCEPTIONS contenant une représentation hexadécimale des caractères non autorisés.
- Continue le traitement de la ligne suivante et ne suit pas l'action d'erreur qui est indiquée pour l'abonnement Q.

La valeur **n** indique à Q Apply de ne pas vérifier les caractères non autorisés et ne signale pas d'exceptions pour ces caractères. Avec cette option, la ligne peut être appliquée à la table cible si DB2 ne rejette pas les données. Si la ligne est appliquée, Q Apply continue le traitement de la ligne suivante. Si les données erronées génèrent une erreur SQL, Q Apply suit l'action d'erreur spécifiée pour l'abonnement Q et signale une exception.

insert_bidi_signal

Valeur par défaut : **insert_bidi_signal=y**

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **insert_bidi_signal** indique si les programmes Q Capture et Q Apply utilisent des insertions de signal pour empêcher la recapture des transactions au cours d'une réplification bidirectionnelle.

Par défaut, le programme Q Apply insère des signaux P2PNORECAPTURE dans la table IBMQREP_SIGNAL pour donner l'instruction au programme Q Capture situé sur le même serveur de ne pas recapter les transactions validées sur ce serveur.

Lorsque les abonnements Q bidirectionnels sont nombreux, le nombre d'insertions de signaux peut affecter les performances de la réplification. Si vous indiquez **insert_bidi_signal=n**, le programme Q Apply n'insère pas de signaux P2PNORECAPTURE. En revanche, vous insérez les informations AUTHTKN de Q Apply dans la table IBMQREP_IGNTRAN, ce qui donne l'instruction au

programme Q Capture situé sur le même serveur de ne pas capturer les transactions issues du programme Q Apply, sauf pour les insertions dans la table IBMQREP_SIGNAL.

Pour améliorer les performances lorsque vous utilisez le paramètre **insert_bidi_signal=n**, mettez à jour la table IBMQREP_IGNTRAN pour définir dans la colonne IGNTRANTRC la valeur N (pas de suivi). Cela évite au programme Q Capture d'insérer une ligne dans la table IBMQREP_IGNTRANTRC pour chaque transaction qu'il ne recapture pas.

loadcopy_path

Valeur par défaut : **loadcopy_path**=Valeur du paramètre *apply_path*

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Utilisation avec la fonction DB2 High Availability Disaster Recovery (HADR) :
Vous pouvez utiliser le paramètre **loadcopy_path** au lieu de la variable de registre DB2_LOAD_COPY_NO_OVERRIDE lorsque le serveur Q Apply est le serveur principal d'une configuration HADR et que les tables du serveur principal sont chargées par le programme Q Apply qui appelle l'utilitaire DB2 LOAD. HADR envoie les fichiers journaux au site secondaire mais, lorsqu'une table du serveur principal est chargée par l'utilitaire DB2 LOAD, les insertions ne sont pas consignées. Si vous indiquez pour le paramètre LOADCOPY_PATH un répertoire NFS accessible à la fois pour le serveur principal et le serveur secondaire, le programme Q Apply démarre l'utilitaire LOAD avec une option qui crée une copie des données chargées dans le chemin spécifié. Le serveur secondaire de la configuration HADR recherche alors les données copiées dans ce chemin.

load_data_buff_sz

Valeur par défaut : **load_data_buff_size**=8

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Utilisation avec des tables de classification multidimensionnelles (MDC) :
Indique le nombre de pages de 4 Ko que l'utilitaire DB2 LOAD doit utiliser comme espace de mémoire tampon pour le transfert des données au sein de l'utilitaire lors du chargement initial de la table cible. Ce paramètre s'applique uniquement aux chargements automatiques à l'aide de l'utilitaire DB2 LOAD.

Par défaut, le programme Q Apply démarre l'utilitaire en indiquant d'utiliser une mémoire tampon de 8 pages : il s'agit également de la valeur minimale qu'il est possible de spécifier pour ce paramètre. En indiquant un nombre de pages plus élevé, il est possible d'améliorer de manière significative les performances de chargement des cibles MDC.

logmarkertz

Valeur par défaut : **logmarkertz**=gmt

Méthodes de modification : au démarrage de Q Apply

Le paramètre **logmarkertz** contrôle le fuseau horaire utilisé par le programme Q Apply lorsqu'il insère des données de validation de la source dans la colonne **IBMSNAP_LOGMARKER** des tables CCD (consistent-change data) ou PIT (point-in-time). Par défaut (**logmarkertz=gmt**), Q Apply insère un horodatage en temps GMT (Greenwich Mean Time) lors de l'enregistrement des données qui ont été validées sur la source. Si les serveurs Q Capture et Q Apply sont situés dans le même fuseau horaire avec une heure d'été (ou autre) identique, vous pouvez indiquer **logmarkertz=local** pour que Q Apply insère un horodatage conforme à l'heure locale du serveur Q Apply.

Remarque : Comme la valeur de la colonne **IBMSNAP_LOGMARKER** enregistre l'heure à laquelle une ligne a été validée dans la base de données source, le paramètre **logmarkertz=local** n'est utile que lorsque les serveurs Q Capture et Q Apply sont situés dans le même fuseau horaire avec une heure d'été (ou autre) identique.

Les lignes existantes dans des cibles CCD ou PIT générées avant l'utilisation de **logmarkertz=local** ne sont pas converties par Q Apply et restent à l'heure GMT, sauf si vous décidez de les modifier manuellement.

Le paramètre **logmarkertz** n'affecte pas les cibles de procédure stockée. Le paramètre **src_commit_timestamp IN** pour les cibles de procédure stockée utilise toujours un horodatage basé sur le temps GMT.

logreuse

Valeur par défaut : **logreuse=n**

Méthodes de modification : au démarrage de Q Apply ; lors de l'exécution de Q Apply ; table **IBMQREP_APPLYPARMS**

Chaque programme Q Apply conserve un fichier journal consignait son historique de travail, par exemple, lorsqu'il commence et cesse la lecture des files d'attente, modifie les valeurs des paramètres, élague les tables de contrôle ou rencontre des erreurs.

Par défaut, le programme Q Apply ajoute une entrée au fichier journal existant lorsqu'il redémarre. Cette valeur par défaut permet de conserver un historique des actions du programme. Si vous ne souhaitez pas conserver cet historique ou si vous désirez économiser de l'espace, définissez le paramètre **logreuse=y**. Le programme Q Apply efface le fichier journal lors de son démarrage, puis écrit dans le fichier vierge.

Le journal est mémorisé par défaut dans le répertoire de démarrage du programme Q Apply ou à un emplacement de votre choix à l'aide du paramètre **apply_path**.

z/OS Le nom du fichier journal est *serveur_apply.schéma_apply.QAPP.1og*. Exemple : *SAMPLE.ASN.APP.1og*. De plus, si le paramètre **apply_path** est défini avec des barres obliques (//) pour utiliser un HLQ (High Level Qualifier), les conventions de noms de fichiers séquentiels z/OS sont appliquées et le paramètre **apply_schema** est tronqué à huit caractères.

Linux UNIX Windows Le nom du fichier journal est *instancedb2.serveur_apply.schéma_apply.QAPP.1og*. Exemple : *DB2.SAMPLE.ASN.QAPP.1og*.

logstdout

Valeur par défaut : `logstdout=n`

Méthodes de modification : au démarrage de Q Apply ; lors de l'exécution de Q Apply ; table IBMQREP_APPLYPARMS

Par défaut, un programme Q Apply écrit uniquement son historique de travail dans le journal. Vous pouvez modifier le paramètre **logstdout** si vous désirez visualiser l'historique du programme sur la sortie standard (stdout), en plus du journal.

Les messages d'erreur et certains messages du journal (initialisation, arrêt, activation et désactivation d'abonnement) sont consignés à la fois dans la sortie standard et dans le fichier journal quelle que soit la valeur de ce paramètre.

Vous pouvez spécifier le paramètre **logstdout** lors du démarrage d'un programme Q Apply avec la commande **asqapp**. Si vous utilisez le Centre de réplication pour démarrer un programme Q Apply, ce paramètre n'est pas applicable.

max_parallel_loads

Valeur par défaut : `max_parallel_loads=1` (z/OS) ; 15 (Linux, UNIX, Windows)

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **max_parallel_loads** indique le nombre maximal d'opérations de chargement automatiques des tables cible que le programme Q Apply peut démarrer en même temps pour une file d'attente de réception donnée. La valeur par défaut pour le paramètre **max_parallel_loads** diffère selon la plateforme du serveur cible :

z/OS

Sous z/OS la valeur par défaut correspond à un chargement à la fois en raison des incidents éventuels provoqués par la procédure stockée DSNUTILS que le programme Q Apply utilise pour appeler l'utilitaire DB2 LOAD. En fonction de votre environnement, vous pouvez essayer de définir une valeur supérieure à **max_parallel_loads=1**. Si des erreurs se produisent, réinitialisez la valeur sur 1.

Linux UNIX Windows

Sous Linux, UNIX et Windows, la valeur par défaut est 15 chargements parallèles.

monitor_interval

Valeurs par défaut : `monitor_interval=60000` millisecondes (1 minute) sous z/OS ; 30000 millisecondes (30 secondes) sous Linux, UNIX et Windows

Méthodes de modification : au démarrage de Q Apply ; lors de l'exécution de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **monitor_interval** indique à un programme Q Apply la fréquence à laquelle il insère des statistiques de performance dans la table

IBMQREP_APPLYMON. Vous pouvez consulter ces statistiques à l'aide des fenêtres Rendement et latence de Q Apply.

Vous pouvez adapter le paramètre `monitor_interval` en fonction de vos besoins :

Si vous souhaitez surveiller les activités d'un programme Q Apply à un niveau plus granulaire, diminuez l'intervalle de contrôle.

Par exemple, vous souhaitez peut-être consulter les statistiques pour connaître le nombre de messages des files d'attente par intervalles d'une 10 secondes au lieu d'une minute.

Augmentez l'intervalle de contrôle pour visualiser les statistiques de performance de Q Apply sur des périodes plus longues

Par exemple, si vous consultez les statistiques de latence pour un grand nombre de périodes d'une minute, il est possible que vous vouliez faire la moyenne des résultats pour obtenir une vue d'ensemble plus large des performances. La visualisation de la moyenne des résultats pour chaque demi-heure ou heure peut s'avérer plus utile dans votre environnement de réplication.

Important pour les utilisateurs du tableau de bord de réplication Q : Lorsque cela est possible, vous devez synchroniser le paramètre `monitor_interval` de Q Apply avec l'intervalle d'actualisation du tableau de bord (à quelle fréquence le tableau de bord récupère des informations de performance des tables de contrôle Q Capture et Q Apply). L'intervalle d'actualisation par défaut du tableau de bord est de 10 secondes (10000 millisecondes). Si la valeur de `monitor_interval` est supérieure à l'intervalle d'actualisation du tableau de bord, ce dernier s'actualise alors qu'aucune nouvelle donnée de surveillance n'est disponible.

monitor_limit

Valeur par défaut : `monitor_limit=10 080` minutes (7 jours)

Méthodes de modification : au démarrage de Q Apply ; lors de l'exécution de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre `monitor_limit` spécifie le degré d'ancienneté que les lignes doivent présenter dans la table IBMQREP_APPLYMON avant de pouvoir être élaguées.

Par défaut, les lignes datant de plus de 10 080 minutes (7 jours) sont élaguées. La table IBMQREP_APPLYMON contient des statistiques sur l'activité d'un programme Q Apply. Une ligne est insérée à chaque intervalle de contrôle. Vous pouvez adapter le nombre maximal de contrôles en fonction de vos besoins :

Augmentez le nombre maximal de contrôles pour conserver plus de statistiques

Si vous souhaitez conserver des enregistrements des activités du programme Q Apply au-delà d'une semaine, définissez un nombre maximal de contrôles supérieur.

Diminuez le nombre maximal de contrôles si vous consultez fréquemment les statistiques.

Si vous surveillez régulièrement les activités du programme Q Apply, vous n'avez probablement pas besoin de conserver les statistiques pendant une semaine complète et pouvez définir un nombre maximal de contrôles inférieur.

Vous pouvez définir le paramètre `monitor_limit` lors du démarrage du programme Q Apply ou pendant son exécution. Vous pouvez également modifier sa valeur

enregistrée dans la table IBMQREP_APPLYPARMS.

nickname_commit_ct

Valeur par défaut : `nickname_commit_ct=10`

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Cibles fédérées : le paramètre `nickname_commit_ct` définit le nombre de lignes après lequel la fonctionnalité IMPORT de DB2 valide les modifications des pseudonymes qui font référence à la table cible pendant le processus de chargement. Ce paramètre s'applique uniquement aux chargements automatiques pour les cibles fédérées, qui emploient l'utilitaire IMPORT.

Par défaut, Q Apply indique si la fonctionnalité IMPORT valide les modifications toutes les 10 lignes pendant le processus de chargement fédéré. Vous devriez voir les performances de chargement s'améliorer en augmentant la valeur du paramètre `nickname_commit_ct`. Par exemple, le paramètre `nickname_commit_ct=100` réduit la charge sur l'UC en limitant les validations intermédiaires. Cependant, une plus grande fréquence de validations protège des incidents qui pourraient se produire pendant la charge, permettant à la fonctionnalité IMPORT d'annuler un nombre moins important de lignes si un incident se produit.

Le paramètre `nickname_commit_ct` est un paramètre de réglage qui permet d'améliorer les performances de la commande DB2 IMPORT en réduisant le nombre de validations pour les cibles fédérées.

p2p_2nodes

Valeur par défaut : `p2p_2nodes=n`

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre `p2p_2nodes` permet au programme Q Apply d'optimiser les performances dans une configuration entre homologues avec deux serveurs actifs uniquement en ne consignant pas les suppressions en conflit dans la table IBMQREP_DELTOMB. N'utilisez le paramètre `p2p_2nodes = y` que pour une réplication entre homologues avec deux serveurs actifs.

Par défaut, le programme Q Apply enregistre les opérations DELETE en conflit dans la table IBMQREP_DELTOMB. Avec le paramètre `p2p_2nodes=y`, le programme Q Apply n'utilise pas la table IBMQREP_DELTOMB. Ceci permet d'éviter les conflits inutiles dans la table ou de ralentir le programme Q Apply sans restreindre la capacité du programme à détecter correctement les conflits et à garantir la convergence des données.

Important : Le programme Q Apply ne détecte pas automatiquement si une configuration entre homologues comporte deux serveurs actifs uniquement. Vérifiez que l'option `p2p_2nodes = y` n'est utilisée que pour une configuration entre homologues avec deux serveurs. Utiliser l'option pour des configurations avec plus de deux serveurs actifs peut engendrer une détection de conflit incorrecte et des divergences de données.

prune_interval

Valeur par défaut : **prune_interval**=300 secondes (5 minutes)

Méthodes de modification : au démarrage de Q Apply ; lors de l'exécution de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **prune_interval** détermine la fréquence à laquelle un programme Q Apply recherche les anciennes lignes à supprimer des tables IBMQREP_APPLYMON et IBMQREP_APPLYTRACE. Par défaut, un programme Q Apply recherche des lignes à élaguer toutes les 300 secondes (5 minutes).

Votre fréquence d'élagage dépend de la rapidité de croissance de vos tables de contrôle et de ce que vous comptez en faire :

Réduisez l'intervalle d'élagage pour gérer des tables de contrôle

Un intervalle d'élagage plus court peut s'avérer nécessaire si la table IBMQREP_APPLYMON croît trop rapidement en raison d'un intervalle de contrôle raccourci. Si cette table n'est pas élaguée suffisamment régulièrement, elle peut dépasser sa limite d'espace, forçant ainsi l'arrêt d'un programme Q Apply. Toutefois, si la table est élaguée trop souvent ou aux heures de pointe, l'élagage peut interférer avec des programmes d'application exécutés sur le même système.

Augmentez l'intervalle d'élagage pour conservation des enregistrements

Il sera peut-être nécessaire de conserver un historique plus long des performances du programme Q Apply en élaguant les tables IBMQREP_APPLYTRACE et IBMQREP_APPLYMON moins régulièrement.

L'intervalle d'élagage fonctionne de pair avec les paramètres **trace_limit** et **monitor_limit**, qui déterminent à partir de quel moment les données sont suffisamment anciennes pour être élaguées. Par exemple, si le paramètre **prune_interval** est de 300 secondes et que le paramètre **trace_limit** est défini sur 10 080 secondes, un programme Q Apply essaiera d'effectuer un élagage toutes les 300 secondes. Si le programme Q Apply détecte des lignes de plus de 10 080 minutes (7 jours) dans la table IBMQREP_APPLYTRACE, il les élague.

Le paramètre **prune_interval** ne s'applique pas à l'élagage de la table IBMQREP_DONEMSG. L'élagage de cette table est contrôlé par les paramètres **prune_method** et **prune_batch_size**.

z/OS

prune_method

Valeur par défaut : **prune_method**=2

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **prune_method** indique la méthode utilisée par le programme Q Apply pour supprimer les lignes inutiles de la table IBMQREP_DONEMSG. Par défaut, (**prune_method**=2), Q Apply élague des groupes de lignes selon la valeur du paramètre **prune_batch_size**. Une unité d'exécution d'élagage distincte enregistre les messages qui ont été appliqués, puis produit une table DELETE basée sur une plage de valeurs.

Lorsque vous indiquez **prune_method=1**, Q Apply élague des lignes de la table IBMQREP_DONESG une par une. Tout d'abord, Q Apply analyse la table pour vérifier si des données provenant d'un message ont été appliquées, supprime le message de la file d'attente de réception, puis élague la ligne correspondante dans IBMQREP_DONEMSG en fournissant une instruction SQL séparée.

pwdfile

Valeur par défaut : **pwdfile=apply_path/asnpwd.aut**

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **pwdfile** indique le nom du fichier de mot de passe codé utilisé par le programme Q Apply pour se connecter au serveur Q Capture. Cette connexion est nécessaire uniquement lorsqu'un abonnement Q spécifie un chargement automatique qui recourt à l'utilitaire EXPORT. Lorsque vous utilisez la commande **asnpwd** pour créer le fichier des mots de passe, le nom de fichier par défaut est **asnpwd.aut**. Si vous attribuez un nom différent au fichier des mots de passe ou si vous modifiez son nom, vous devez alors modifier le paramètre **pwdfile** en conséquence. Le programme Q Apply recherche le fichier des mots de passe dans le répertoire spécifié par le paramètre **apply_path**.

z/OS

Aucun mot de passe n'est requis.

Vous pouvez définir le paramètre **pwdfile** lors du démarrage du programme Q Apply et modifier sa valeur enregistrée dans la table IBMQREP_APPLYPARMS. Vous ne pouvez pas modifier la valeur pendant l'exécution du programme Q Apply.

report_exception

Valeur par défaut : **report_exception=y**

Méthodes de modification : au démarrage de Q Apply

Le paramètre **report_exception** détermine si le programme Q Apply insère des données dans la table IBMQREP_EXCEPTIONS en cas de conflit ou d'erreur SQL au niveau de la table cible, sauf que la ligne est appliquée de toute façon à la cible, car l'action liée au conflit et définie pour l'abonnement Q était F (forcer). Par défaut, (**report_exception=y**), Q Apply insère des informations dans la table IBMQREP_EXCEPTIONS pour chacune des lignes ayant provoqué un conflit ou une erreur SQL sur la cible, que la ligne ait été appliquée ou non. Vous pouvez définir **report_exception=n** pour que Q Apply n'insère pas de données dans la table IBMQREP_EXCEPTIONS si une ligne provoque un conflit, tout en étant appliquée. Avec **report_exception=n**, Q Apply continue à insérer des données concernant des lignes qui n'ont pas été appliquées.

Lorsque **report_exception=n**, le programme Q Apply tolère également des erreurs de conversion de page de code lors de l'écriture de texte SQL dans la table IBMQREP_EXCEPTIONS et continue le traitement normal.

richklvl

Valeur par défaut : **richklvl=2**

Méthodes de modification : au démarrage de Q Apply ; table
IBMQREP_APPLYPARMS

Le paramètre **richklvl** indique le niveau de vérification de l'intégrité référentielle. Par défaut (**richklvl**=2), le programme Q Apply vérifie les dépendances basées sur RI entre les transactions, afin de s'assurer que les lignes dépendantes sont appliquées dans le bon ordre.

Si vous spécifiez **richklvl**=5, Q Apply vérifie les dépendances basées sur RI lorsqu'une valeur clé est mise à jour dans la table parent, une ligne est mise à jour dans la table parent ou une ligne est supprimée de la table parent.

La valeur 0 indique à Q Apply de ne pas vérifier les dépendances basées sur RI.

Lorsqu'une transaction ne peut pas être appliquée en raison de la violation de l'intégrité référentielle, le programme Q Apply réessaye automatiquement la transaction jusqu'à ce qu'elle soit appliquée dans le même ordre dans lequel elle a été validée dans la table source.

spill_commit_count

Valeur par défaut : **spill_commit_count**=10

Méthodes de modification : au démarrage de Q Apply ; table
IBMQREP_APPLYPARMS

Le paramètre **spill_commit_count** indique le nombre de lignes regroupées dans une portée de validation par les agents de déversement Q Apply qui appliquent les données répliquées dans le cadre de l'opération de chargement. En augmentant le nombre de lignes appliquées avant l'exécution d'une commande COMMIT, vous pouvez améliorer les performances en limitant les ressources E/S nécessaires pour des validations fréquentes. Il convient de trouver un compromis entre l'amélioration des performances et les risques de conflit d'accès à la table cible et à la table de contrôle IBMQREP_SPILLEDROW si les validations sont moins nombreuses.

skiptrans

Valeur par défaut : aucune

Méthode de modification : au démarrage de Q Apply

Le paramètre **skiptrans** indique que le programme Q Apply ne doit pas appliquer une ou plusieurs transactions d'une ou plusieurs files d'attente de réception, selon leur ID de transaction.

Il peut être utile d'empêcher le traitement de certaines transactions par le programme dans certaines situations imprévues, comme par exemple :

- Le programme Q Apply reçoit une erreur lors de l'application d'une ligne d'une transaction, et s'arrête complètement ou arrête de lire cette file d'attente de réception. Au redémarrage, vous souhaitez peut-être que le programme Q Apply ignore l'ensemble de la transaction qui a provoqué l'erreur.
- Après un basculement dans le cadre d'une reprise après incident, vous préférez peut-être ignorer une plage de transactions de la file d'attente de réception, depuis le noeud défaillant vers le noeud de secours.

Vous pouvez également demander au programme Q Capture d'ignorer des transactions. Cela s'avère utile dans les cas où vous pouvez planifier les transactions qui n'ont pas besoin d'être répliquées.

Remarque : Toutefois, si vous ignorez une transaction qui avait été validée sur le serveur source, vous risquez d'avoir des divergences entre les tables source et cible. Dans ce cas, vous pouvez utiliser les utilitaires **asntdiff** et **asntrep** pour synchroniser les tables.

startallq

Valeur par défaut : **startallq=n** (z/OS) ; y (Linux, UNIX, Windows)

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **startallq** indique comment Q Apply traite les files d'attente de réception lorsqu'il démarre. Lorsque **startallq=y**, Q Apply place toutes les files d'attente de réception à l'état actif et commence à les lire lorsqu'il démarre. Lorsque vous indiquez **startallq=n**, Q Apply traite uniquement les files d'attente de réception actives au démarrage.

Vous pouvez utiliser **startallq=y** pour ne pas avoir à émettre la commande **startq** pour les files d'attente de réception inactives après le démarrage du programme Q Apply. Vous pouvez utiliser **startallq=n** pour que les files d'attente désactivées restent inactives au démarrage de Q Apply.

term

Valeur par défaut : **term=y**

Méthodes de modification : au démarrage de Q Apply ; lors de l'exécution de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **term** détermine si un programme Q Apply continue de s'exécuter lorsque DB2 ou le gestionnaire de files d'attente est indisponible.

Par défaut (**term=y**), le programme Q Apply s'arrête lorsque DB2 ou le gestionnaire de files d'attente n'est pas disponible. Vous pouvez modifier la valeur par défaut (**term=n**) si vous souhaitez que le programme Q Apply poursuive son exécution pendant que DB2 ou le gestionnaire de files d'attente est indisponible. Lorsque DB2 ou le gestionnaire de files d'attente est disponible, Q Apply commence à appliquer des transactions là où il s'est arrêté sans qu'il soit nécessaire de redémarrer le programme.

Remarque : Quelle que soit la valeur du paramètre **term**, si les canaux émetteurs ou récepteurs WebSphere MQ s'arrêtent, le programme Q Apply continue de s'exécuter car il ne peut pas détecter le statut des canaux. Cette situation provoque l'arrêt de la réplication car les deux gestionnaires de files d'attente ne peuvent pas communiquer. Si la réplication a été arrêtée sans aucun message des programmes de réplication, recherchez des erreurs WebSphere MQ. Par exemple, vérifiez le statut des canaux à l'aide de la commande WebSphere MQ DISPLAY CHSTATUS.

trace_ddl

Valeur par défaut : **trace_ddl=n**

Méthodes de modification : au démarrage de Q Apply ; table IBMQREP_APPLYPARMS

Lorsque des opérations DDL effectuées au niveau de la base de données source sont répliquées, le paramètre **trace_ddl** indique si le texte SQL de l'opération que le programme Q Apply exécute au niveau de la base de données cible est consigné. Par défaut (**trace_ddl=n**), Q Apply ne consigne pas le texte SQL. Si vous indiquez **trace_ddl=y**, Q Apply envoie un message ASN dans le fichier journal de Q Apply, émet une sortie standard et produit une table IBMQREP_APPLYTRACE avec le texte de l'instruction SQL. Le texte SQL est tronqué à 1024 caractères.

trace_limit

Valeur par défaut : **trace_limit=10 080** minutes (7 jours)

Méthodes de modification : au démarrage de Q Apply ; lors de l'exécution de Q Apply ; table IBMQREP_APPLYPARMS

Le paramètre **trace_limit** indique la durée de conservation des lignes longues dans la table IBMQREP_APPLYTRACE avant de pouvoir être élaguées.

Le programme Q Apply insère tous les messages d'information, d'avertissement et d'erreur dans la table IBMQREP_APPLYTRACE. Par défaut, les lignes de plus de 10 080 minutes (7 jours) sont élaguées à chaque intervalle d'élagage. Modifiez la limite de trace en fonction de vos besoins pour des informations d'audit.

Exemples d'utilisation de la commande asnqapp

Ces exemples illustrent comment utiliser la commande **asnqcap**.

Spécification d'un chemin d'accès aux fichiers de travail et au fichier de mot de passe

Pour lancer un programme Q Apply sur le serveur targetdb avec le schéma alpha, des fichiers de travail situés dans le répertoire /home/files/qapply, et l'utilisation du fichier de mot de passe pass1.txt :

```
asnqapp apply_server=targetdb apply_schema="alpha"  
apply_path="/home/files/qapply" pwdfile="pass1.txt"
```

Raccourcissement de l'intervalle de surveillance pour qu'il corresponde au tableau de bord de réplication Q

Pour lancer un programme Q Apply et raccourcir l'intervalle d'insertion de lignes dans la table IBMQREP_APPLYMON de la valeur par défaut de 60000 millisecondes (60 secondes) sur z/OS à 10000 millisecondes (10 secondes) afin qu'il corresponde à l'intervalle d'actualisation par défaut du tableau de bord de réplication Q, procédez comme suit :

```
asnqapp apply_server=targetdb apply_schema="alpha" monitor_interval=10000
```

Arrêt du programme Q Apply une fois toutes les files d'attente vidées

Pour lancer un programme Q Apply et arrêter son fonctionnement une fois que toutes les files d'attente ont été vidées une fois :

```
asnqapp apply_server=targetdb apply_schema="alpha" autostop=y
```

Arrêt de Q Apply avec un horodatage spécifié

Pour démarrer un programme Q Apply et lui donner l'instruction de s'arrêter de fonctionner après avoir traité des transactions validées à la source jusqu'à un horodatage spécifié :

```
asnqapp apply_server=targetdb apply_schema="alpha" applyupto="2011-04-02-20"
```

Cet exemple utilise le format d'horodatage partiel pour arrêter le programme Q Apply lorsqu'il lit une transaction avec un horodatage situé après le 2 avril 2011 à 20 heures en temps GMT (Greenwich Mean Time).

Démarrage et arrêt du programme Q Apply par instruction

Pour démarrer un programme Q Apply et lui donner l'instruction de s'arrêter de fonctionner après avoir lu un horodatage correspondant à l'heure à laquelle vous avez démarré Q Apply, ou lorsque la file d'attente de réception est vide (quel que soit l'événement qui se produit en premier) :

```
asnqapp apply_server=targetdb apply_schema="alpha"  
applyupto=CURRENT_TIMESTAMP,NOWAIT
```

Dire à Q Apply d'ignorer les transactions spécifiées

Pour démarrer un programme Q Apply et indiquer qu'il ne doit pas appliquer une transaction donnée de la file d'attente de réception Q1 et une plage de transactions de la file d'attente de réception Q2 :

```
asnqapp apply_server=targetdb apply_schema=ASN  
skiptrans="Q1;0000:0000:0000:0000:51a1,  
Q2;0000:0000:0000:0000:51b0-0000:0000:0000:0000:51c0"
```

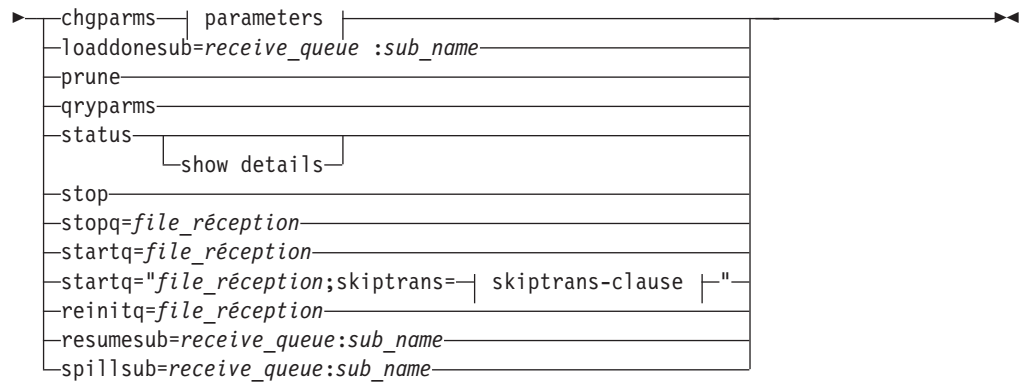
asnqacmd : utilisation d'un programme Q Apply en cours d'exécution

Utilisez la commande **asnqacmd** avec les services système Linux, UNIX, Windows, et UNIX sous z/OS pour envoyer une commande à un programme Q Apply en cours de fonctionnement. Exécutez cette commande à l'invite du système d'exploitation ou dans un script de shell.

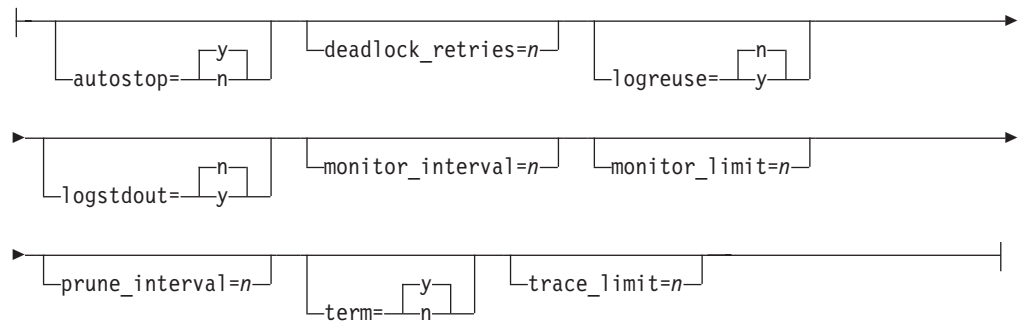
Pour plus d'informations sur l'utilisation de la commande MVS MODIFY pour envoyer des commandes à un programme Q Capture en cours d'exécution sous z/OS, voir Gestion des programmes de réplication Q et de publication d'événements en cours d'exécution à l'aide de la commande MVS MODIFY.

Syntaxe

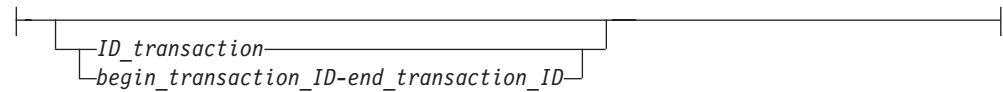
```
▶▶ asnqacmd—apply_server=nom_bd—┬───┬───▶  
└───┬───▶  
    └───▶ apply_schema=schéma ───▶
```

Paramètres :



skiptrans-clause ::



Paramètres

Le tableau 42 définit les paramètres d'appel applicables à la commande **asnqacmd**.

Tableau 42. Définitions des paramètres d'appel de la commande **asnqacmd**

Paramètre	Définition
apply_server=nom_bd	Indique le nom de la base de données ou du sous-système contenant les tables de contrôle Q Apply.
	<p>z/OS Spécifie le nom du sous-système DB2 sur lequel le programme Q Apply s'exécutera. Pour le partage de données, utilisez le nom de liaison de groupe, de façon à exécuter Q Apply dans n'importe quel LPAR sans modifier le langage JCL de la tâche démarrée.</p> <p>Linux UNIX Windows Si vous n'indiquez pas de serveur Q Apply, ce paramètre prend par défaut la valeur de la variable d'environnement DB2DBDFT.</p>
apply_schema=schéma	Indique un nom qui identifie le programme Q Apply en cours de fonctionnement à utiliser.

Tableau 42. Définitions des paramètres d'appel de la commande *asqacmd* (suite)

Paramètre	Définition
chgparms	<p>Spécifie le changement d'un ou de plusieurs paramètres opérationnels parmi les paramètres suivants du programme Q Apply pendant le fonctionnement de ce dernier :</p> <ul style="list-style-type: none"> • autostop • deadlock_retries • • • logreuse • logstdout • monitor_interval • monitor_limit • prune_interval • term • trace_limit <p>Important : Les paramètres modifiés doivent suivre immédiatement le paramètre chgparms.</p> <p>Vous pouvez indiquer plusieurs paramètres avec commande chgparms et modifier leur valeur aussi souvent que vous le souhaitez. Les changements apportés remplacent temporairement les valeurs de la table IBMQREP_APPLYPARMS, mais elles ne sont pas enregistrées dans cette table. Lorsque vous arrêtez puis redémarrez le programme Q Apply, ce dernier utilise les valeurs affectées à IBMQREP_APPLYPARMS. Pour plus d'informations sur les paramètres qui peuvent être remplacés avec cette commande, voir «Descriptions des paramètres <i>asqapp</i>», à la page 417.</p>
loaddonesub= <i>receive_queue:sub_name</i>	<p>Signaler d'informer le programme Q Apply qu'un chargement manuel de la table cible pour l'abonnement Q est terminé. N'utilisez ce paramètre que si un chargement manuel est spécifié pour l'abonnement Q (la colonne HAS_LOADPHASE dans la table IBMQREP_SUBS a la valeur de E). Lorsque Q Apply reçoit la commande, il commence par traiter les messages dans la file d'attente auxiliaire pour l'abonnement Q. Lorsque tous les messages auxiliaires sont traités, Q Apply change l'état de l'abonnement Q en A (actif).</p> <p>Réplication bidirectionnelle ou entre homologues : Lancez cette commande pour le même abonnement Q que vous avez démarré pour commencer le processus d'activation de groupe. Cet abonnement Q est aussi celui dont la table source a été utilisée comme source de chargement.</p>
prune	<p>Utilisez ce paramètre pour indiquer au programme Q Apply qu'il doit supprimer les anciennes entrées des tables IBMQREP_APPLYMON et IBMQREP_APPLYTRACE. Cet élagage vient compléter les élagages réguliers planifiés conformément aux indications du paramètre prune_interval.</p>
qryparms	<p>Indiquez si vous souhaitez que les valeurs actuelles des paramètres opérationnels du programme Q Apply soient enregistrées dans la sortie standard (stdout).</p>
status	<p>Précisez pour voir un message sur l'état de chaque unité d'exécution Q Apply (principal, gestion interne, surveillance, navigateurs et agents).</p>

Tableau 42. Définitions des paramètres d'appel de la commande *asnlqcmd* (suite)

Paramètre	Définition
show details	<p>Indiquez après le paramètre status si vous souhaitez afficher un rapport plus détaillé sur l'état du programme Q Apply, contenant les informations suivantes sur l'instance Q Apply :</p> <ul style="list-style-type: none"> • Programme Q Apply en cours de fonctionnement ou non • Temps écoulé depuis le démarrage du programme • Emplacement du journal de diagnostic du programme Q Apply • Nombre d'abonnements Q actifs • Période utilisée pour le calcul des valeurs moyennes <p>Les informations suivantes s'affichent également pour chaque file d'attente de réception active :</p> <ul style="list-style-type: none"> • Nom de la file d'attente • Nombre d'abonnements Q actifs • Toutes les transactions appliquées depuis (heure) (OLDEST_TRANS) • Point de reprise pour Q Capture (MAXCMTSEQ) • Toutes les transactions appliquées depuis (LSN) • Transaction en cours la plus ancienne (OLDEST_INFLT_TRANS) • temps d'attente de bout-en-bout • temps d'attente du programme Q Capture • temps d'attente de la file d'attente • temps d'attente du programme Q Apply • Quantité de mémoire (exprimée en octets) utilisée par l'unité d'exécution du navigateur pour la lecture des transactions à partir de la file d'attente • Nombre de messages présents dans la file d'attente (longueur de file d'attente) • Pourcentage de remplissage de la file d'attente • Unités d'exécution d'agent qui traitent des transactions • Unités d'exécution d'agent qui attendent des transactions • Unités d'exécution d'agent qui traitent des messages internes • Agents en cours d'initialisation
stop	Indiquez si vous souhaitez arrêter le programme Q Apply et valider les messages traités jusqu'à ce point.
stopq=file_réception	Utilisez ce paramètre pour indiquer au programme Q Apply qu'il doit arrêter le traitement des messages pour une file d'attente de réception donnée. Toutes les transactions en mémoire sont traitées.
startq=file_réception	Indiquez ce paramètre pour demander au programme Q Apply de démarrer le traitement des messages dans une file d'attente de réception.
startq= "file_réception; skiptrans= ID_transaction"	Utilisez ce paramètre pour indiquer au programme Q Apply qu'il ne doit pas appliquer une transaction ou une plage de transactions d'une file d'attente de réception donnée lorsque vous démarrez le traitement des messages pour la file d'attente de réception. Pour plus de détails sur la spécification d'un identificateur de transaction ou d'une plage d'identificateurs, voir la section relative au paramètre skiptrans dans «asnlqpp : lancement d'un programme Q Apply», à la page 416.
reinitq=file_réception	Utilisez ce paramètre pour indiquer au programme Q Apply qu'il doit mettre à jour les attributs de NUM_APPLY_AGENTS et de MEMORY_LIMIT issus de la table IBMQREP_RECVQUEUEES, pour tous les abonnements Q qui utilisent une file d'attente de réception spécifique. Cette commande fonctionne uniquement si le programme Q Apply effectue la lecture à partir de la file d'attente de réception nommée dans la table IBMQREP_RECVQUEUEES lorsque vous émettez la commande.
resumesub=file_réception:sous-nom	<p>Utilisez ce paramètre pour reprendre l'application des lignes déversées sur la table cible. Spécifiez le nom de la file d'attente de réception et de l'abonnement Q identifiant la table cible. Le paramètre sp11sub est utilisé pour placer l'abonnement Q dans un état de déversement (S) et le paramètre resumesub est utilisé pour reprendre les opérations normales.</p> <p>Les agents de déversement appliquent les lignes à partir de la file d'attente auxiliaire temporaire jusqu'à ce que la file d'attente soit vide. Lorsque les agents de déversement traitent les lignes, les lignes entrantes sont ajoutées à la file d'attente auxiliaire. Lorsque la file d'attente auxiliaire est vidée, l'état de l'abonnement Q est défini sur Actif (A) et les opérations normales reprennent.</p>

Tableau 42. Définitions des paramètres d'appel de la commande `asnlqcmd` (suite)

Paramètre	Définition
<code>spillsub=file_réception:sous-nom</code>	Utilisez ce paramètre pour que toutes les modifications apportées à une ligne pour un abonnement Q soient redirigées vers une file d'attente auxiliaire temporaire. Vous pouvez effectuer des opérations de maintenance sur la table cible comme la réorganisation de la table. Spécifiez le nom de la file d'attente de réception et de l'abonnement Q identifiant la table cible. La file d'attente auxiliaire est créée en fonction de la définition de votre file d'attente modèle. Assurez-vous que la longueur maximale de la file d'attente est suffisamment grande pour conserver les lignes déversées jusqu'à ce qu'elles puissent être appliquées après la reprise des opérations.

Exemple 1

Pour mettre à jour tous les abonnements Q qui utilisent la file d'attente Q1 avec les dernières valeurs de nombre d'agents Q Apply et de limite de mémoire (issues de la table `IBMQREP_RECVQUEUES`) :

```
asnlqcmd apply_server=targetdb apply_schema="alpha" reinitq=Q1
```

Exemple 2

Pour demander à un programme Q Apply en cours d'exécution de s'arrêter après le vidage de toutes les files d'attente, et de réduire l'intervalle du moniteur et la limite de trace :

```
asnlqcmd apply_server=targetdb apply_schema="alpha" chgparms autostop=y  
monitor_interval=60000 trace_limit=5000
```

Exemple 3

Pour recevoir des messages relatifs à l'état de chaque unité d'exécution Q Apply :

```
asnlqcmd apply_server=targetdb apply_schema="alpha" status
```

Exemple 4

Pour recevoir des messages détaillés sur le programme Q Apply et sur les files d'attente de réception actives :

```
asnlqcmd apply_server=targetdb apply_schema="alpha" status show details
```

Exemple 5

Pour supprimer les anciennes entrées des tables `IBMQREP_APPLYMON` et `IBMQREP_APPLYTRACE` :

```
asnlqcmd apply_server=targetdb apply_schema="alpha" prune
```

Exemple 6

Pour placer un abonnement Q en mode déversement pour effectuer des opérations de maintenance sur la table cible :

```
asnlqcmd apply_server=targetdb apply_schema="BSN"  
spillsub="BSN.QM1_TO_QM2.recvq:EmployeeSub"
```

Une fois les opérations de maintenance terminées sur la table cible, utilisez le paramètre `resumesub` pour reprendre les opérations normales.

Exemple 7

Pour demander au programme Q Apply d'ignorer une plage de transactions lorsqu'il commence à traiter la file de réception Q2 :

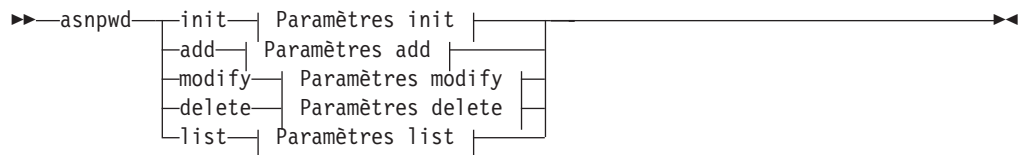
```
asnqcmd apply_server=targetdb apply_schema=ASN  
startq="Q2;skiptrans=0000:0000:0000:0000:51a1-0000:0000:0000:0000:51a8"
```

asnpwd : création et gestion des fichiers de mots de passe

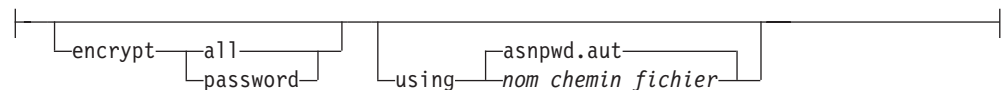
Utilisez la commande **asnpwd** pour créer et modifier les fichiers sous Linux, UNIX et Windows. Exécutez cette commande sur la ligne de commande ou dans un script de shell.

L'aide relative à la commande apparaît si vous saisissez la commande **asnpwd** sans aucun paramètre, suivi d'un point d'interrogation (?) ou de paramètres incorrects.

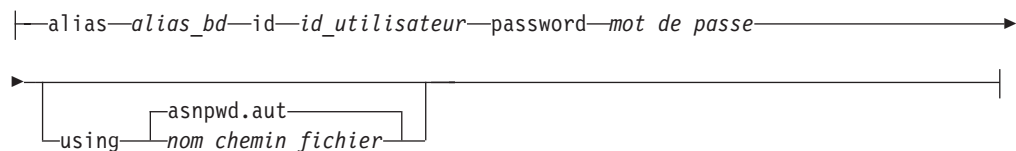
Syntaxe



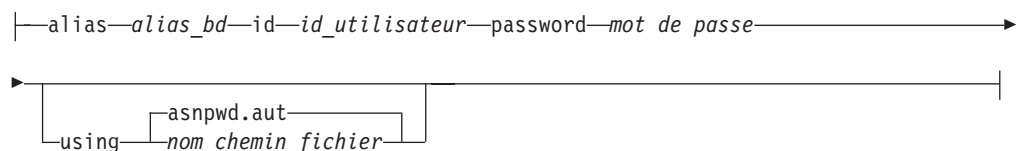
Paramètres init :



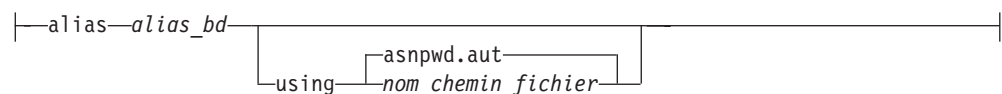
Paramètres add :



Paramètres modify :



Paramètres delete :



Paramètres list :



Paramètres

Le tableau 43 définit les paramètres d'appel de la commande **asnpwd**.

Remarque importante concernant la compatibilité des fichiers de mots de passe : A partir de la version 9.5 avec le groupe de correctifs 2, les fichiers de mots de passe qui sont créés par la commande **asnpwd** utilisent une nouvelle méthode de chiffrement et ne peuvent pas être lus par des versions plus anciennes des programmes et des utilitaires de réplication. Si vous partagez un fichier de mot de passe parmi des programmes et des utilitaires au niveau mixte avec des groupes de correctifs plus anciens, ne recréez pas le fichier de mot de passe à l'aide d'un utilitaire **asnpwd** faisant partie de ces groupes de correctifs ou de groupes plus récents. Les programmes et utilitaires de réplication de ces groupes de correctifs ou de groupes plus récents peuvent toujours fonctionner avec des fichiers de mots de passe plus anciens. De plus, il est impossible de modifier un fichier de mot de passe plus ancien pour utiliser la nouvelle méthode de chiffrement. Vous devez créer un nouveau fichier de mot de passe.

Remarque concernant l'utilisation : Dans les systèmes d'exploitation Windows 64 bits, les options ADD, MODIFY, DELETE et LIST ne sont pas prises en charge pour les fichiers de mots de passe qui ont été créés à l'aide de la commande **asnpwd** avant la version 9.5, groupe de correctifs 2.

Tableau 43. Définitions des paramètres d'appel de la commande **asnpwd** pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Définition
init	Indiquez ce paramètre pour créer un fichier de mot de passe vide. Cette commande échouera si vous indiquez le paramètre init avec un fichier de mot de passe déjà existant.
add	Indiquez ce paramètre pour ajouter une entrée au fichier de mot de passe. Dans le fichier de mot de passe, il ne peut y avoir qu'une seule entrée par <i>alias_bd</i> . Cette commande échouera si vous indiquez le paramètre add avec une entrée déjà existante dans le fichier de mot de passe. Utilisez le paramètre modify pour modifier une entrée existante dans le fichier de mot de passe.
modify	Indiquez ce paramètre pour modifier le mot de passe ou l'ID utilisateur d'une entrée dans le fichier de mot de passe.
delete	Indiquez ce paramètre pour supprimer une entrée du fichier de mot de passe.
list	Indiquez ce paramètre pour répertorier les alias et les entrées d'ID utilisateur dans un fichier de mot de passe. Ce paramètre ne peut être utilisé que si le fichier de mots de passe a été créé à l'aide du paramètre encrypt password . Les mots de passe ne sont jamais affichés par la commande list .

Tableau 43. Définitions des paramètres d'appel de la commande `asnpwd` pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Définition
encrypt	Indique les entrées d'un fichier qui doivent être chiffrées. all (valeur par défaut) Chiffre toutes les entrées dans le fichier spécifié de sorte que vous ne pouvez pas lister les alias de la base de données, les noms d'utilisateurs et les mots de passe qui figurent dans le fichier. Cette option réduit l'exposition des informations dans les fichiers de mots de passe. password Chiffre l'entrée du mot de passe dans le fichier spécifié. Cette option permet aux utilisateurs de répertorier les alias de la base de données et les noms d'utilisateurs mémorisés dans leur fichier de mot de passe. Les mots de passe ne peuvent jamais être affichés.
using <i>chemin_fichier</i>	Indique le chemin d'accès et le nom du fichier de mot de passe. Respectez les conventions d'attribution de noms de votre système d'exploitation. <code>C:\sql11ib\mypwd.aut</code> est un exemple de fichier de mot de passe valide sous Windows. Si vous spécifiez le chemin d'accès et le nom du fichier de mot de passe, le chemin d'accès et le fichier de mot de passe doivent déjà exister. Si vous utilisez le paramètre init et que vous indiquez le chemin et le nom du fichier de mot de passe, le chemin d'accès doit déjà exister et la commande créera le fichier de mot de passe à votre place. Si vous n'indiquez pas ce paramètre, le nom de fichier par défaut est <code>asnpwd.aut</code> et le chemin d'accès au fichier par défaut est le répertoire actuel.
alias <i>alias_bd</i>	Indique l'alias de la base de données à laquelle l'ID utilisateur a accès. L'alias est toujours compacté en majuscules, quelle que soit la façon dont il est saisi.
id <i>id_utilisateur</i>	Indique l'ID utilisateur qui a accès à la base de données.
password <i>mot de passe</i>	Indique le mot de passe de l'ID utilisateur fourni. Ce mot de passe est sensible à la casse et est chiffré dans le fichier de mot de passe.

Codes retour

La commande **asnpwd** renvoie un code retour égal à zéro une fois l'opération réussie. Un code retour autre que zéro est renvoyé si la commande ne s'est pas exécutée normalement.

Exemples de commande `asnpwd`

Les exemples suivants illustrent comment utiliser la commande **asnpwd**.

Exemple 1

Pour créer un fichier de mot de passe avec le nom par défaut `asnpwd.aut` dans le répertoire actuel :

```
asnpwd INIT
```

Exemple 2

Pour créer un fichier de mot de passe appelé `pass1.aut` dans le répertoire `c:\myfiles` :

```
asnpwd INIT USING c:\myfiles\pass1.aut
```

Exemple 3

Pour créer un fichier de mots de passe appelé `mypwd.aut` avec le paramètre **encrypt all** :

```
asnpwd INIT ENCRYPT ALL USING mypwd.aut
```

Exemple 4

Pour créer un fichier de mots de passe appelé `mypwd.aut` avec le paramètre **encrypt password** :

```
asnpwd INIT ENCRYPT PASSWORD USING mypwd.aut
```

Exemple 5

Pour créer un fichier de mots de passe par défaut à l'aide du paramètre **encrypt password** :

```
asnpwd INIT ENCRYPT PASSWORD
```

Exemple 6

Pour ajouter un ID utilisateur nommé `oneuser` et son mot de passe au fichier de mot de passe appelé `pass1.aut` dans le répertoire `c:\myfiles`, et lui octroyer un accès à la base de données `db1` :

```
asnpwd ADD ALIAS db1 ID oneuser PASSWORD mypwd using c:\myfiles\pass1.aut
```

Exemple 7

Pour modifier l'ID utilisateur ou le mot de passe d'une entrée dans le fichier de mot de passe appelé `pass1.aut` dans le répertoire `c:\myfiles` :

```
asnpwd MODIFY AliaS sample ID chglocalid PASSWORD chgmajorpwd  
USING c:\myfiles\pass1.aut
```

Exemple 8

Pour supprimer l'alias de la base de données appelé `sample` du fichier de mot de passe nommé `pass1.aut` dans le répertoire `c:\myfiles` :

```
asnpwd delete alias sample USING c:\myfiles\pass1.aut
```

Exemple 9

Pour afficher l'aide de la commande :

```
asnpwd
```

Exemple 10

Pour répertorier les entrées dans un fichier de mot de passe par défaut :

```
asnpwd LIST
```

Exemple 11

Pour répertorier les entrées dans un fichier de mot de passe appelé pass1.aut :

```
asnpwd LIST USING pass1.aut
```

La sortie de cette commande dépend de la manière dont le fichier a été initialisé :

- Si le fichier a été initialisé avec le paramètre **encrypt all**, le message suivant s'affiche :

```
ASN1986E "Asnpwd" : "". Le fichier de mot de passe "pass1.aut"
contient des informations
chiffrées impossibles à répertorier.
```

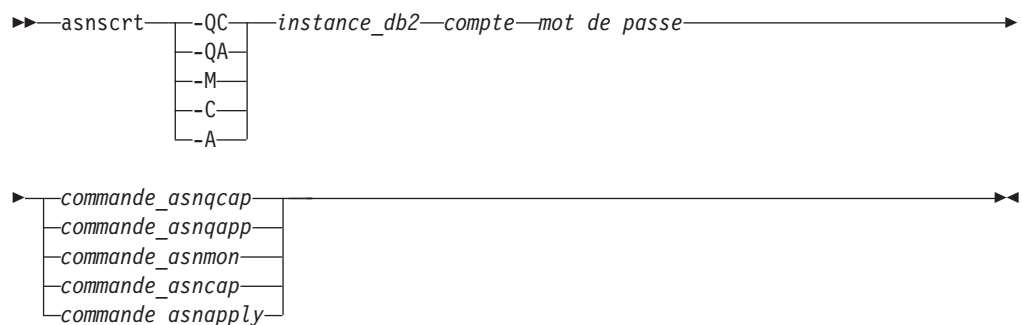
- Si le fichier n'a pas été initialisé avec le paramètre **encrypt all**, les détails suivants sont indiqués :

```
asnpwd LIST USING pass1.aut
Alias : SAMPLE ID : chglocalid
Nombre d'entrées : 1
```

asnsCRT : création d'un service de réplication

La commande **asnsCRT** permet de créer un service de réplication dans le Gestionnaire de contrôle de service (SCM) Windows et d'appeler les commandes **asnsqcap**, **asnsqapp**, **asnsmon**, **asnsqcap** et **asnsapply**. Exécutez la commande **asnsCRT** sur le système d'exploitation Windows.

Syntaxe



Paramètres

Le tableau 44 définit les paramètres d'appel de la commande **asnsCRT**.

Tableau 44. définition des paramètres d'appel de la commande **asnsCRT** pour les systèmes d'exploitation Windows

Paramètre	Définition
-QC	Indique que vous lancez un programme Q Capture.
-QA	Indique que vous lancez un programme Q Apply.
-M	Indique que vous lancez un programme Moniteur d'alertes de réplication.
-C	Indique que vous lancez un programme Capture.
-A	Indique que vous lancez un programme Apply.
db2_instance	Indique l'instance DB2 utilisée pour identifier un service de réplication DB2 unique. L'instance DB2 peut contenir jusqu'à huit caractères.

Tableau 44. définition des paramètres d'appel de la commande *asnsrct* pour les systèmes d'exploitation Windows (suite)

Paramètre	Définition
account	Indique le nom de compte que vous utilisez pour vous connecter à Windows. Si le compte est local, il doit commencer par un point et une barre oblique inverse (. \). Autrement, le nom de domaine ou de l'ordinateur doit être spécifié (par exemple, <code>nom_domaine\nom_compte</code>).
password	Indique le mot de passe utilisé avec le nom de compte. Si le mot de passe contient des caractères spéciaux, saisissez une barre oblique inverse (\) avant chaque caractère spécial.
asnqcap_command	<p>Indique la commande asnqcap complète pour lancer un programme Q capture. Utilisez la syntaxe documentée de la commande asnqcap avec les paramètres asnqcap.</p> <p>Si la variable d'environnement DB2PATH n'est pas définie, vous devez indiquer un emplacement pour les fichiers de travail en incluant le paramètre capture_path via la commande asnqcap. Si la variable DB2PATH est définie et que vous indiquez un paramètre capture_path, le paramètre capture_path remplace la variable DB2PATH.</p> <p>La commande asnsrct ne valide pas la syntaxe des paramètres asnqcap saisis.</p>
asnqapp_command	<p>Indique la commande asnqapp complète pour lancer un programme Q apply. Utilisez la syntaxe documentée de la commande asnqapp avec les paramètres asnqapp appropriés.</p> <p>Si la variable d'environnement DB2PATH n'est pas définie, vous devez indiquer l'emplacement pour les fichiers de travail en incluant le paramètre apply_path via la commande asnqapp. Si la variable DB2PATH est définie et que vous indiquez un paramètre apply_path, le paramètre apply_path remplace la variable DB2PATH. La commande asnsrct ne valide pas la syntaxe des paramètres asnqapp saisis.</p>
asnmon_command	<p>Indique la commande asnmon complète pour lancer un programme Moniteur d'alertes de réplication. Utilisez la syntaxe documentée de la commande asnmon avec les paramètres asnmon.</p> <p>Si la variable d'environnement DB2PATH n'est pas définie, vous devez indiquer un emplacement pour les fichiers journaux en incluant le paramètre monitor_path via la commande asnmon. Si la variable DB2PATH est définie et que vous indiquez un paramètre chemin_monitor, ce dernier remplace la variable DB2PATH.</p> <p>La commande asnsrct ne valide pas la syntaxe des paramètres asnmon saisis.</p>

Tableau 44. définition des paramètres d'appel de la commande `asnscri` pour les systèmes d'exploitation Windows (suite)

Paramètre	Définition
<code>asncap_command</code>	<p>Indique la commande asncap complète pour lancer un programme Capture. Utilisez la syntaxe documentée de la commande asncap avec les paramètres asncap appropriés.</p> <p>Si la variable d'environnement DB2PATH n'est pas définie, vous devez indiquer un emplacement pour les fichiers de travail en incluant le paramètre capture_path via la commande asncap. Si la variable DB2PATH est définie et que vous indiquez un paramètre capture_path, le paramètre capture_path remplace la variable DB2PATH.</p> <p>La commande asnscri ne valide pas la syntaxe des paramètres asncap saisis.</p>
<code>asnapply_command</code>	<p>Indique la commande asnapply complète pour lancer un programme Apply. Utilisez la syntaxe documentée de la commande asnapply avec les paramètres asnapply appropriés.</p> <p>Si la variable d'environnement DB2PATH n'est pas définie, vous devez indiquer l'emplacement pour les fichiers journaux en incluant le paramètre apply_path via la commande asnapply. Si la variable DB2PATH est définie et que vous indiquez un paramètre apply_path, le paramètre apply_path remplace la variable DB2PATH.</p> <p>La commande asnscri ne valide pas la syntaxe des paramètres asnapply saisis.</p>

Exemples de commande `asnscri`

Les exemples suivants illustrent comment utiliser la commande **asnscri**.

Exemple 1

Pour créer un service de réplication DB2 qui appelle un programme Q Apply sous une instance DB2 appelée `inst2` à l'aide du compte de connexion `.\joesmith` et du mot de passe `my$pwd` :

```
asnscri -QA inst2 .\joesmith my\$pwd asnqapp apply_server=mydb2 apply_schema =as2
  apply_path=X:\sqllib
```

Exemple 2

Pour créer un service de réplication DB2 qui appelle un programme Capture sous une instance DB2 appelée `inst1` :

```
asnscri -C inst1 .\joesmith password asncap capture_server=sampled
  capture_schema=ASN capture_path=X:\logfiles
```

Exemple 3

Pour créer un service de réplication DB2 qui appelle un programme Apply sous une instance DB2 appelée `inst2` à l'aide du compte de connexion `.\joesmith` et du mot de passe `my$pwd` :

```
asnscri -A inst2 .\joesmith my\$pwd asnapply control_server=db2 apply_qual=aq2
  apply_path=X:\sqllib
```

Exemple 4

Pour créer un service de réplication DB2 qui appelle un programme Moniteur d'alertes de réplication sous une instance DB2 appelée inst3 :

```
asnscri -M inst3 .\joesmith password asnmon monitor_server=db3 monitor_qual=mq3  
monitor_path=X:\logfiles
```

Exemple 5

Pour créer un service de réplication DB2 qui appelle un programme Capture sous une instance DB2 appelée inst4 et remplace le répertoire des fichiers de travail par défaut par un paramètre **capture_path** qualifié complet :

```
asnscri -C inst4 .\joesmith password X:\sql1ib\bin\asnca capture_server=scdb  
capture_schema=ASN capture_path=X:\logfiles
```

Exemple 6

Pour créer un service de réplication DB2 qui appelle un programme Q capture sous une instance DB2 appelée inst1 :

```
asnscri -QC inst1 .\joesmith password asnqcap capture_server=mydb1  
capture_schema=QC1 capture_path=X:\logfiles
```

asnsdrop : suppression d'un service de réplication

La commande **asnsdrop** permet de supprimer les services de réplication à partir du Gestionnaire de contrôle de service (SCM) Windows sur le système d'exploitation Windows.

Syntaxe

```
▶▶ asnsdrop [ nom_service ]
```

Le diagramme ci-dessus illustre la syntaxe de la commande **asnsdrop**. Elle est suivie d'un paramètre facultatif **nom_service**, qui peut être remplacé par **ALL**. Des flèches indiquent que le paramètre peut être omis.

Paramètres

Le tableau 45 définit les paramètres d'appel de la commande **asnsdrop**.

Tableau 45. définitions des paramètres d'appel de la commande asnsdrop pour les systèmes d'exploitation Windows

Paramètre	Définition
nom_service	Spécifie le nom complet du service de réplication DB2. Accédez au Gestionnaire de contrôle de service (SCM) Windows pour obtenir le nom du service de réplication DB2. Sur les systèmes d'exploitation Windows, vous pouvez obtenir le nom du service en ouvrant la fenêtre Propriétés du service de réplication DB2. Si le nom du service de réplication DB2 contient des espaces, entourez le nom complet du service de guillemets doubles.
ALL	Indique que vous souhaitez déplacer tous les services de réplication DB2.

Exemples de commande asnsdrop

Les exemples suivants illustrent comment utiliser la commande **asnsdrop**.

Exemple 1

Pour déplacer un service de réplication DB2 :

```
asnsdrop DB2.SAMPLEDB.SAMPLEDB.CAP.ASN
```

Exemple 2

Pour déplacer un service de réplication DB2 avec un schéma nommé A S N (contenant des blancs imbriqués), entourez le nom du service de guillemets doubles :

```
asnsdrop "DB2.SAMPLEDB.SAMPLEDB.CAP.A S N"
```

Exemple 3

Pour déplacer tous les services de réplication DB2 :

```
asnsdrop ALL
```

asnslist : liste des services de réplication

La commande **asnslist** permet de répertorier les services de réplication dans le Gestionnaire de contrôle de service (SCM) Windows. Vous pouvez, en option, utiliser la commande pour lister les détails de chaque service. Exécutez la commande **asnslist** sur le système d'exploitation Windows.

Syntaxe

```
▶▶ asnslist [DETAILS] ▶▶
```

Paramètres

Le tableau 46 définit le paramètre d'appel de la commande **asnslist**.

Tableau 46. définition du paramètre d'appel de la commande asnslist pour les systèmes d'exploitation Windows

Paramètre	Définition
details	Indique que vous souhaitez lister des données détaillées de tous les services de réplication DB2 sur un système.

Exemples de la commande asnslist

Les exemples suivants illustrent comment utiliser la commande **asnslist**.

Exemple 1

Pour lister les noms des services de réplication DB2 sur un système :

```
asnslist
```

Voici un exemple de sortie de ces informations :

```
DB2.DB2.SAMPLE.QAPP.ASN
DB2.DB4.SAMPLE.QCAP.ASN
```

Exemple 2

Pour lister les détails de tous les services sur un système :

```
asnslist details
```

Voici un exemple de sortie de ces informations :

```
DB2.DB2.SAMPLE.QAPP.ASN
Nom affiché : DB2 DB2 SAMPLE QAPPLY ASN
Chemin d'accès à l'image : ASNSERV DB2.DB2.SAMPLE.APP.AQ1
-ASNQAPPLY QAPPLY_SERVER=SAMPLE AP
          PLY_SCHEMA=ASN QAPPLY_PATH=C:\PROGRA~1\SQLLIB
Dépendance : DB2-0

DB2.DB4.SAMPLE.QCAP.ASN
Nom affiché : DB2 DB4 SAMPLE QAPPLY ASN
Chemin d'accès à l'image : ASNSERV DB2.DB4.SAMPLE.APP.AQ1
-ASNQCAP QCAPTURE_SERVER=SAMPLE CA
          PTURE_SCHEMA=ASN QCAPTURE_PATH=C:\PROGRA~1\SQLLIB
Dépendance : DB4-0
```

asntdiff : comparaison des données dans les tables source et cible (Linux, UNIX, Windows)

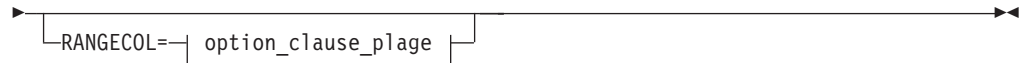
La commande **asntdiff** permet de comparer deux tables relationnelles et de générer une liste des différences entre les deux. Exécutez la commande **asntdiff** à l'invite du système d'exploitation ou dans un script de shell.

La présente rubrique décrit l'utilisation de la commande sous Linux, UNIX ou Windows. Pour plus de détails sur l'exécution de **asntdiff** sous z/OS, voir «asntdiff : comparaison des données dans les tables source et cible (z/OS)», à la page 454. Pour obtenir des informations sur l'option de commande **asntdiff -f** qui permet de comparer des tables ne faisant pas partie de la réplication à l'aide d'un fichier en entrée, voir «Option de commande **asntdiff -f** (fichier en entrée)», à la page 461.

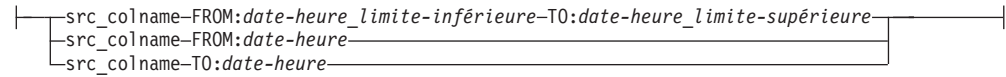
Les tables que vous comparez peuvent se trouver sous DB2 for Linux, UNIX, Windows, DB2 for z/OS ou DB2 for System i.

Syntaxe

```
▶▶ asntdiff—DB=serveur—SCHEMA=schéma—
  —DIFF_SCHEMA=schéma_table_différences—DIFF_TABLESPACE=espace_table—
  —DIFF_DROP=n—y—MAXDIFF=limite_différence—WHERE=clause_WHERE—
▶ —DIFF_PATH=chemin_journal—PWFILe=nom_fichier—DIFF=nom_table—
```



option_clause_plage :



Paramètres

Le tableau 47 définit les paramètres d'appel de la commande **asntdiff**.

Tableau 47. Définitions des paramètres d'appel de la commande **asntdiff** pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Définition
DB = <i>serveur</i>	Indique l'alias DB2 de la base de données contenant des informations sur les tables source et cible à comparer. La valeur diffère selon que vous utilisez la réplication Q ou la réplication SQL : Réplication Q Nom du serveur Q Capture qui contient la table IBMQREP_SUBS. Réplication SQL Nom du serveur de contrôle Apply qui contient la table IBMSNAP_SUBS_MEMBR.
SCHEMA = <i>schéma</i>	Indique le schéma des tables de contrôle Q Capture pour la réplication Q ou le schéma des tables de contrôle Apply pour la réplication SQL. Il s'agit par défaut de ASN.
DIFF_SCHEMA = <i>schéma_table_différences</i>	Indique le schéma de la table des différences. Il s'agit par défaut de ASN.
DIFF_TABLESPACE = <i>espace_table</i>	Indique l'espace table de la table de la table des différences. Si ce paramètre n'est pas spécifié, la table est créée dans l'espace table par défaut de la base de données dans laquelle la commande asntdiff a été exécutée.
DIFF_DROP = <i>y/n</i>	Indique si une table de différences existante sera supprimée et recréée avant d'être réutilisée pour enregistrer les différences. Si la table n'existe pas, la commande asntdiff en crée une. n (par défaut) La table des différences sera utilisée en l'état et les lignes existantes seront supprimées. y La table des différences sera supprimée et recréée.
MAXDIFF = <i>limite_différence</i>	Indique le nombre maximal de différences que la commande asntdiff doit traiter avant de s'arrêter. La valeur par défaut est 10000.

Tableau 47. Définitions des paramètres d'appel de la commande *asntdiff* pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Définition
WHERE = <i>clause_WHERE</i>	<p>Indique une clause WHERE SQL qui identifie de manière unique une ligne de la table de contrôle contenant des informations sur les tables source et cible à comparer. La clause WHERE doit être entourée de guillemets doubles. La valeur de ce paramètre diffère selon que vous utilisez la réplication Q ou la réplication SQL :</p> <p>Réplication Q La clause WHERE indique une ligne de la table IBMQREP_SUBS en utilisant la colonne SUBNAME pour identifier l'abonnement Q qui contient les tables source et cible.</p> <p>Réplication SQL La clause WHERE indique une ligne de la table IBMSNAP_SUBS_MEMBR en utilisant les colonnes SET_NAME, APPLY_QUAL, TARGET_SCHEMA et TARGET_TABLE pour identifier le membre de l'ensemble d'abonnements qui contient les tables source et cible.</p>
DIFF_PATH = <i>chemin_journal</i>	<p>Indique l'emplacement dans lequel la commande <i>asntdiff</i> doit écrire son journal. La valeur par défaut est le répertoire depuis lequel vous avez exécuté la commande. La valeur doit être un nom de chemin absolu. Pour conserver la casse, utilisez des guillemets doubles ("").</p>
PWDFILE = <i>nom_fichier</i>	<p>Indique le nom du fichier de mot de passe qui permet de se connecter aux bases de données. Si vous ne spécifiez aucun fichier de mot de passe, la valeur par défaut est <i>asnpwd.aut</i> (nom du fichier de mot de passe créé par la commande <i>asnpwd</i>). La commande <i>asntdiff</i> recherche le fichier de mot de passe dans le répertoire qui est spécifié par le paramètre DIFF_PATH. Si aucune valeur n'est spécifiée pour le paramètre DIFF_PATH, la commande recherche le fichier de mot de passe dans le répertoire où la commande a été exécutée.</p>
DIFF = <i>nom_table</i>	<p>Indique le nom de la table créée dans la base de données source pour enregistrer les différences entre les tables source et cible. La table comporte une ligne pour chaque différence détectée. Si vous n'ajoutez pas ce paramètre ou le paramètre DIFF_SCHEMA, la table des différences s'appelle ASN.ASNTRDIFF.</p>

Tableau 47. Définitions des paramètres d'appel de la commande `asntdiff` pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Définition
Clause RANGECOL	<p>Indique une plage de lignes de la table source que vous souhaitez comparer. Vous indiquez le nom d'une colonne DATE, TIME ou TIMESTAMP dans la table source, puis utilisez l'une des trois clauses pour spécifier la plage. Le nom de la colonne doit être entouré de guillemets simples. La clause doit être entourée de guillemets doubles.</p> <p>L'horodatage utilise le format suivant : <i>AAAA-MM-JJ-HH.MM.SS.mmmmm</i>. Par exemple, 2010-03-10-10.35.30.55555 est l'horodatage GMT pour le 10 mars 2010, 10h35, 30 secondes et 55 555 microsecondes.</p> <p>Utilisez l'une des clauses suivantes :</p> <p><i>nomcol_src</i> FROM: <i>date-heure</i> limite-inférieure TO: <i>date-heure</i> limite-supérieure</p> <p>Indique une limite inférieure et supérieure pour la plage de lignes à comparer.</p> <p>L'exemple suivant illustre une colonne HORODATAGE :</p> <pre>" 'SALETIME' FROM: 2008-02-08-03.00.00.00000 TO: 2008-02-15-03.00.00.00000"</pre> <p>A faire : Les mots clés FROM: et TO: sont obligatoires et doivent tous deux être suivis de deux points (:).</p> <p><i>nomcol_src</i> FROM: <i>date-heure</i></p> <p>Indique que vous souhaitez comparer toutes les lignes contenant des horodatages supérieurs ou égaux à <i>date-heure</i>.</p> <p>Par exemple :</p> <pre>" 'SALE_TIME' FROM: 2008-03-10-10.35.30.55555"</pre> <p><i>nomcol_src</i> TO: <i>date-heure</i></p> <p>Indique que vous souhaitez comparer toutes les lignes contenant des horodatages inférieurs ou égaux à <i>date-heure</i>.</p> <p>Par exemple :</p> <pre>" 'SALETIME' TO: 2008-03-20-12.00.00.00000"</pre> <p>Recommandation : Pour des performances optimales, assurez-vous qu'un index de la colonne source est spécifié dans la clause <code>plage</code>. Lorsque vous comparez des tables impliquées dans la réplication entre homologues, vous pouvez utiliser la colonne <code>IBMQREPVERTIME</code> générée par IBM pour la colonne source dans la clause <code>plage</code>.</p> <p>Restriction : Le paramètre <code>RANGECOL</code> n'est pas valide pour l'option <code>asntdiff -f</code> (fichier en entrée). Vous pouvez utiliser une clause <code>WHERE SQL</code> dans le fichier en entrée pour obtenir des résultats similaires.</p>

Exemples de commande `asntdiff`

Les exemples suivants illustrent comment utiliser la commande `asntdiff`.

Exemple 1

Dans la réplication Q, pour identifier les différences entre une table source et cible qui sont spécifiées dans un abonnement Q appelé my_qsub, sur un serveur Q Capture nommé source_db, avec le schéma Q Capture asn :

```
asntdiff db=source_db schema=asn where="subname = 'my_qsub'"
```

Exemple 2

Dans la réplication SQL, pour identifier les différences entre une table source et cible qui sont spécifiées dans un ensemble d'abonnements appelé my_set, avec une table cible nommée trg_table, sur un serveur de contrôle Apply appelé apply_db, avec le schéma Apply asn, et pour attribuer un nom à la table des différences diff_table :

```
asntdiff DB=apply_db schema=asn where="set_name = 'my_set'
and target_table = 'trg_table'" diff=diff_table
```

Exemple 3

Dans la réplication Q, pour identifier les différences entre une plage de lignes dans les tables source et cible qui sont indiquées dans un abonnement Q entre homologues appelé my_qsub, sur un serveur Q Capture nommé source_db, avec le schéma Q Capture asn :

```
asntdiff db=source_db schema=asn where="subname = 'my_qsub'"
RANGECOL="IBMQRPEPVERTIME' FROM: '2008-03-10-0.00.00.000000'
TO: '2007-04-12-00.00.00.000000'"
```

Exemple 4

Dans la réplication SQL, pour identifier les différences entre une plage de lignes dans les tables source et cible qui sont indiquées dans un ensemble d'abonnements appelé my_set, avec une table cible nommée trg_table, sur un serveur de contrôle Apply appelé apply_db, avec le schéma Apply asn, et pour attribuer un nom à la table de différences diff_table :

```
asntdiff DB=apply_db schema=asn where="set_name = 'my_set'
and target_table = 'trg_table'" diff=diff_table
RANGECOL = "'CREDIT_TIME' FROM: '2008-03-10-12.00.00.000000'
TO: '2008-03-11-12.00.00.000000'"
```

asntdiff : comparaison des données dans les tables source et cible (z/OS)

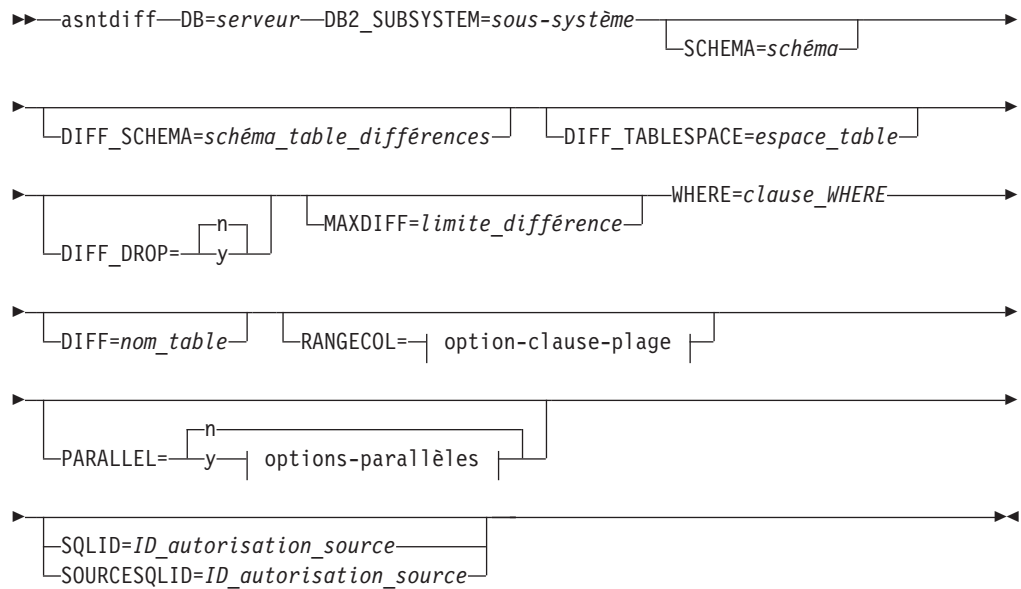
La commande **asntdiff** permet de comparer deux tables relationnelles et de générer une liste des différences entre les deux. Exécutez la commande **asntdiff** avec JCL ou dans une invite de commande des services système UNIX (USS) ou dans un script de shell.

La présente rubrique décrit l'utilisation sous z/OS :

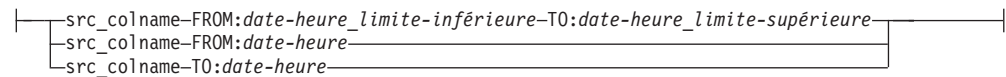
- Pour plus de détails sur l'exécution de **asntdiff** sous Linux, UNIX et Windows, voir «asntdiff : comparaison des données dans les tables source et cible (Linux, UNIX, Windows)», à la page 450.
- Pour plus d'informations sur l'option de commande asntdiff -f, qui vous permet de comparer des tables qu'elles soient impliquées ou non dans la réplication, voir «Option de commande asntdiff -f (fichier en entrée)», à la page 461.
- Pour plus de détails quant à l'utilisation de asntdiff en mode parallèle, voir «Exécution de l'utilitaire asntdiff en mode parallèle (sous z/OS)», à la page 347.

Les tables que vous comparez peuvent résider sur DB2 for z/OS, DB2 pour Linux, UNIX, Windows ou DB2 pour System i.

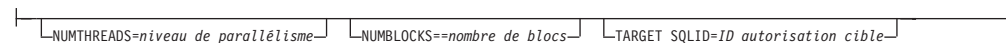
Syntaxe



option-clause-plage :



options-parallèles :



Paramètres

Le tableau 48, à la page 456 définit les paramètres d'appel de la commande `asntdiff`.

Tableau 48. Définitions des paramètres d'appel de la commande `asntdiff` pour les systèmes d'exploitation z/OS

Paramètre	Définition
<code>DB=serveur</code>	<p>Indique l'alias DB2 de la base de données contenant des informations sur les tables source et cible à comparer. La valeur diffère selon que vous utilisez la réplication Q ou la réplication SQL :</p> <p>Réplication Q Nom de l'emplacement du serveur Q Capture qui contient la table <code>IBMQREP_SUBS</code>.</p> <p>Réplication SQL Nom de l'emplacement du serveur de contrôle Apply qui contient la table <code>IBMSNAP_SUBS_MEMBR</code>.</p> <p>Restriction : Ce paramètre n'est pas valide pour l'option <code>asntdiff -f</code> (fichier en entrée). Vous pouvez utiliser les paramètres <code>SOURCE_SERVER</code> et <code>TARGET_SERVER</code> avec l'option <code>-f</code> afin de spécifier les noms des bases de données source et cible. Sous z/OS, il s'agit de noms d'emplacements et vous devez également utiliser le paramètre <code>DB2_SUBSYSTEM</code> pour spécifier le nom du sous-système où est exécuté l'utilitaire <code>asntdiff</code>.</p>
<code>DB2_SUBSYSTEM=sous-système</code>	Indique le nom du sous-système dans lequel vous exécutez la commande <code>asntdiff</code> .
<code>SCHEMA=schéma</code>	Indique le schéma des tables de contrôle Q Capture pour la réplication Q ou le schéma des tables de contrôle Apply pour la réplication SQL. Il s'agit par défaut de <code>ASN</code> .
<code>DIFF_SCHEMA=schéma_table_différences</code>	Indique le schéma de la table des différences. Il s'agit par défaut de <code>ASN</code> .
<code>DIFF_TABLESPACE=espace_table</code>	<p>Indique l'espace table de la table de la table des différences. Si ce paramètre n'est pas spécifié, la table est créée dans l'espace table par défaut du sous-système dans lequel la commande <code>asntdiff</code> a été exécutée.</p> <p>Ce nom comprend deux parties : <code>nombd.espace_table</code>, où <code>nombd</code> correspond au nom logique de la base de données et <code>espace_table</code> au nom de l'espace table.</p>
<code>DIFF_DROP=y/n</code>	<p>Indique si une table de différences existante sera supprimée et recréée avant d'être réutilisée pour enregistrer les différences. Si la table n'existe pas, la commande <code>asntdiff</code> en crée une.</p> <p>n (par défaut) La table des différences sera utilisée en l'état et les lignes existantes seront supprimées.</p> <p>y La table des différences sera supprimée et recréée.</p>
<code>MAXDIFF=limite_différence</code>	Indique le nombre maximal de différences que la commande <code>asntdiff</code> doit traiter avant de s'arrêter. La valeur par défaut est 10000.

Tableau 48. Définitions des paramètres d'appel de la commande *asntdiff* pour les systèmes d'exploitation z/OS (suite)

Paramètre	Définition
WHERE = <i>clause_WHERE</i>	<p>Indique une clause WHERE SQL qui identifie de manière unique une ligne de la table de contrôle contenant des informations sur les tables source et cible à comparer. La clause WHERE doit être entourée de guillemets doubles. La valeur de ce paramètre diffère selon que vous utilisez la réplication Q ou la réplication SQL :</p> <p>Réplication Q La clause WHERE indique une ligne de la table IBMQREP_SUBS en utilisant la colonne SUBNAME pour identifier l'abonnement Q qui contient les tables source et cible.</p> <p>Réplication SQL La clause WHERE indique une ligne de la table IBMSNAP_SUBS_MEMBR en utilisant les colonnes SET_NAME, APPLY_QUAL, TARGET_SCHEMA et TARGET_TABLE pour identifier le membre de l'ensemble d'abonnements qui contient les tables source et cible.</p> <p>Restriction : Ce paramètre n'est pas valide pour l'option <i>asntdiff -f</i> (fichier en entrée). Vous pouvez utiliser les paramètres SOURCE_SELECT et TARGET_SELECT avec l'option <i>-f</i> afin de spécifier les tables à comparer et pouvez indiquer une clause WHERE dans la requête SQL fournie avec ces paramètres.</p>
DIFF = <i>nom_table</i>	<p>Indique le nom de la table créée dans le sous-système source pour enregistrer les différences entre les tables source et cible. La table comporte une ligne pour chaque différence détectée. Si vous n'ajoutez pas ce paramètre ou le paramètre DIFF_SCHEMA, la table des différences s'appelle ASN.ASNTDIFF.</p>

Tableau 48. Définitions des paramètres d'appel de la commande *asntdiff* pour les systèmes d'exploitation z/OS (suite)

Paramètre	Définition
Clause RANGECOL	<p>Indique une plage de lignes de la table source que vous souhaitez comparer. Vous indiquez le nom d'une colonne DATE, TIME ou TIMESTAMP dans la table source, puis utilisez l'une des trois clauses pour spécifier la plage. Le nom de la colonne doit être entouré de guillemets simples. La clause doit être entourée de guillemets doubles.</p> <p>L'horodatage utilise le format suivant : <i>AAAA-MM-JJ-HH.MM.SS.mmmmm</i>. Par exemple, 2010-03-10-10.35.30.55555 est l'horodatage GMT pour le 10 mars 2010, 10h35, 30 secondes et 55 555 microsecondes.</p> <p>Utilisez l'une des clauses suivantes :</p> <p><i>nomcol_src</i> FROM: <i>date-heure</i> limite-inférieure TO: <i>date-heure</i> limite-supérieure</p> <p>Indique une limite inférieure et supérieure pour la plage de lignes à comparer.</p> <p>L'exemple suivant illustre une colonne HORODATAGE :</p> <pre>" 'SALETIME' FROM: 2008-02-08-03.00.00.00000 TO: 2008-02-15-03.00.00.00000"</pre> <p>A faire : Les mots clés FROM: et TO: sont obligatoires et doivent tous deux être suivis de deux points (:).</p> <p><i>nomcol_src</i> FROM: <i>date-heure</i></p> <p>Indique que vous souhaitez comparer toutes les lignes contenant des horodatages supérieurs ou égaux à <i>date-heure</i>.</p> <p>Par exemple :</p> <pre>" 'SALE_TIME' FROM: 2008-03-10-10.35.30.55555"</pre> <p><i>nomcol_src</i> TO: <i>date-heure</i></p> <p>Indique que vous souhaitez comparer toutes les lignes contenant des horodatages inférieurs ou égaux à <i>date-heure</i>.</p> <p>Par exemple :</p> <pre>" 'SALETIME' TO: 2008-03-20-12.00.00.00000"</pre> <p>Recommandation : Pour des performances optimales, assurez-vous qu'un index de la colonne source est spécifié dans la clause <i>plage</i>. Lorsque vous comparez des tables impliquées dans la réplication entre homologues, vous pouvez utiliser la colonne IBMQREPVERTIME générée par IBM pour la colonne source dans la clause <i>plage</i>.</p> <p>Restriction : Le paramètre RANGECOL n'est pas valide pour l'option <i>asntdiff -f</i> (fichier en entrée). Vous pouvez utiliser une clause WHERE SQL dans le fichier en entrée pour obtenir des résultats similaires.</p>

Tableau 48. Définitions des paramètres d'appel de la commande **asntdiff** pour les systèmes d'exploitation z/OS (suite)

Paramètre	Définition
PARALLEL =y/n	<p>Spécifie si l'utilitaire asntdiff utilise le mode parallèle, dans lequel plusieurs unités d'exécution sont utilisées pour la comparaison des tables, ou s'il opère en série avec une unité d'exécution unique.</p> <p>n (par défaut) L'utilitaire asntdiff utilise le mode en série.</p> <p>y L'utilitaire asntdiff utilise le mode parallèle. Pour plus de détails sur les exigences d'installation, les autorisations obligatoires et les restrictions, voir «Exécution de l'utilitaire asntdiff en mode parallèle (sous z/OS)», à la page 347.</p>
NUMTHREADS = nombre_d'unités_d'exécution	<p>Spécifie le nombre d'unités d'exécution que l'utilitaire asntdiff est autorisé à créer. La valeur minimale est six. La valeur recommandée est 21, qui correspond par ailleurs à la valeur maximale et à la valeur par défaut. Assurez-vous que la valeur du paramètre MAXTHREADS spécifiée dans BPXPRMXX soit supérieure au nombre spécifié d'unités d'exécution. En outre, configurez DB2 ZPARMS CTHREAD, IDFORE et IDBACK pour autoriser chacune des unités d'exécution créées à se connecter à DB2.</p>
NUMBLOCKS =nombre_de_blocs	<p>Spécifie le nombre de partitions selon lequel l'utilitaire asntdiff divise les tables source et cible (c'est-à-dire, les ensembles de résultats des paramètres SOURCE_SELECT et TARGET_SELECT) pour comparaison parallèle. Une valeur de 0 (par défaut) signifie que l'utilitaire détermine automatiquement le nombre de blocs.</p>
SQLID =ID_autorisation	<p>Utilisez ce paramètre lorsque asntdiff est exécuté en mode non-parallèle. Le paramètre spécifie un identificateur d'autorisation pouvant être utilisé pour créer la table de différence. Utilisez ce paramètre si l'identificateur permettant d'exécuter la commande asntdiff n'est pas autorisé à créer des tables. La valeur du paramètre SQLID est utilisée en tant que schéma pour la table des différences si vous ne spécifiez pas explicitement un schéma à l'aide du paramètre DIFF_SCHEMA.</p>
SOURCE_SQLID =ID_autorisation	<p>Lorsque vous utilisez asntdiff en mode parallèle, ce paramètre spécifie un identificateur d'autorisation pouvant être utilisé pour exécuter des procédures et des packs mémorisés et exécuter des langages de définition de données et des bibliothèques de supports définitifs sur les tables temporaires à la source. Utilisez ce paramètre si l'identificateur utilisé pour exécuter la commande asntdiff ne dispose pas de l'autorisation nécessaire.</p>
TARGET_SQLID =ID_autorisation	<p>Lorsque vous utilisez asntdiff en mode parallèle, ce paramètre spécifie un identificateur d'autorisation pouvant être utilisé pour exécuter des procédures et des packs mémorisés et exécuter des langages de définition de données et des bibliothèques de supports définitifs sur les tables temporaires à la cible. Utilisez ce paramètre si l'identificateur utilisé pour exécuter la commande asntdiff ne dispose pas de l'autorisation nécessaire.</p>

Notes sur l'utilisation

La commande **asntdiff** crée des ensembles de données (JCL) ou des fichiers temporaires (USS) pour déverser des données et enregistrer les différences avant de les insérer dans la table des différences. L'emplacement des ensembles de données ou des fichiers temporaires doit être indiqué de manière distincte :

JCL

Si vous voulez que la commande ASNTDIFF effectue des enregistrements sur des ensembles de données z/OS, ajoutez les deux instructions DD suivantes à votre ASNTDIFF JCL, en modifiant les spécifications de taille afin de les faire correspondre à la taille de votre table source :

```
//SPLFILE DD DSN=&&SPILL,DISP=(NEW,DELETE,DELETE),
//          UNIT=VIO,SPACE=(CYL,(11,7)),
//          DCB=(RECFM=VS,BLKSIZE=6404)
//DIFFFILE DD DSN=&&DIFFLE,DISP=(NEW,DELETE,DELETE),
//          UNIT=VIO,SPACE=(CYL,(11,7)),
//          DCB=(RECFM=VS,BLKSIZE=6404)
```

USS Sous USS, les fichiers temporaires sont enregistrés par défaut dans le système hiérarchique de fichiers (HFS), dans le répertoire de base de l'ID utilisateur qui exécute la commande **asntdiff**. Les noms par défaut sont DD:DIFFFILE et DD:SPILLFILE. Vous pouvez utiliser une instruction DIFFFILE DD pour spécifier un autre chemin HFS et un autre nom de fichier pour ces fichiers, comme illustré dans l'exemple ci-dessous :

```
//DIFFFILE DD PATH='/u/oeusr01/tdiffi12',
//          PATHDISP=(KEEP,KEEP),
//          PATHOPTS=(ORDWR,OCREAT),
//          PATHMODE=(SIRWXU,SIRGRP,SIROTH)
```

La redirection du système hiérarchique de fichiers implique la création d'un fichier vide vers le quel les enregistrements peuvent être effectués ou l'utilisation des paramètres PATHDISP et PATHOPTS pour créer un nouveau fichier s'il n'existe pas.

Exemples de commande asntdiff

Les quatre premiers exemples illustrent la procédure d'utilisation de la commande **asntdiff** sous USS. Le cinquième exemple fournit le langage JCL. Pour plus de modèles JCL, reportez-vous au modèle de programme ASNTDIFF dans le modèle de fichier SASNSAMP.

Exemple 1 : réplication Q

Dans la réplication Q, pour identifier les différences entre une table source et cible qui sont spécifiées dans un abonnement Q appelé my_qsub, sur un serveur Q Capture nommé source_db, avec le schéma Q Capture asn :

```
asntdiff db=source_db schema=asn where="subname = 'my_qsub'"
```

Exemple 2 : réplication SQL

Dans la réplication SQL, pour identifier les différences entre une table source et cible qui sont spécifiées dans un ensemble d'abonnements appelé my_set, avec une table cible nommée trg_table, sur un serveur de contrôle Apply appelé apply_db, avec le schéma Apply asn, et pour attribuer un nom à la table des différences diff_table :

```
asntdiff DB=apply_db schema=asn where="set_name = 'my_set'
and target_table = 'trg_table'" diff=diff_table
```

Exemple 3 : comparaison d'une plage de lignes dans la réplication Q

Dans la réplication Q, pour identifier les différences entre une plage de lignes dans les tables source et cible qui sont indiquées dans un abonnement Q entre homologues appelé my_qsub, sur un serveur Q Capture nommé source_db, avec le schéma Q Capture asn :

```
asntdiff db=source_db schema=asn where="subname = 'my_qsub'"
RANGECOL="'IBMQRÉPVERTIME' FROM: '2008-03-10-0.00.00.00000'"
TO: '2007-04-12-00.00.00.00000'"
```

Exemple 4 : comparaison d'une plage de lignes dans la réplication SQL

Dans la réplication SQL, pour identifier les différences entre une plage de lignes dans les tables source et cible qui sont indiquées dans un ensemble d'abonnements appelé my_set, avec une table cible nommée trg_table, sur un serveur de contrôle Apply appelé apply_db, avec le schéma Apply asn, et pour attribuer un nom à la table de différences diff_table :

```
asntdiff DB=apply_db schema=asn where="set_name = 'my_set'
and target_table = 'trg_table'" diff=diff_table
RANGECOL = "'CREDIT_TIME' FROM: '2008-03-10-12.00.00.00000'"
TO: '2008-03-11-12.00.00.00000'"
```

Exemple 5 : utilisation de asntdiff en mode parallèle

Pour exécuter l'utilitaire asntdiff en mode parallèle afin de comparer deux tables avec 21 unités d'exécution parallèles, vous pouvez utiliser le JCL suivant après avoir localisé et modifié toutes les occurrences des chaînes suivantes :

- DSN! du sous-système pour le nom de votre sous-système DB2
- DSN!!0 pour le nom de votre bibliothèque destinataire DB2
- ASNQ!!0 pour le nom de la bibliothèque destinataire de votre serveur de réplication

```
//ASNTDIF1 EXEC PGM=ASNTDIFF,PARM='/-F'
//STEPLIB DD DSN=ASNQ!!0.SASNLOAD,DISP=SHR
// DD DSN=DSN!!0.SDSNLOAD,DISP=SHR
//MSGSD DD PATH='/usr/lpp/db2rep1_10_01/msg/En_US/db2asn.cat'
//CEEDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD DUMMY
//SYSIN DD *
DB2_SUBSYSTEM=DSN!
SOURCE_SERVER=DQRG
SOURCE_SELECT="SELECT empno, department FROM employee
WHERE empno > 10000 ORDER BY 1"
TARGET_SERVER=D7DP
TARGET_SELECT="SELECT empno, department FROM employee
WHERE empno > 10000 ORDER BY 1"
PARALLEL=Y
NUMTHREADS=21
SOURCE_SQLID=SRCADM
TARGET_SQLID=TGTADM
DIFF_DROP=Y
MAXDIFF=20000
DEBUG=NO
/*
//
```

Option de commande asntdiff -f (fichier en entrée)

Avec l'option de commande **asntdiff -f**, vous utilisez un fichier en entrée pour spécifier des informations sur n'importe laquelle des deux tables que vous voulez comparer qu'elles soient répliquées ou non.

Le fichier en entrée contient les instructions SQL SELECT pour les tables source et cible qui spécifient les lignes à comparer. La commande `asntdiff` standard compare les tables qui sont impliquées dans la réplication en utilisant des informations d'abonnement à partir des tables de contrôle de réplication.

L'option **asntdiff -f** permet de comparer n'importe quelle table z/OS, Linux, UNIX ou Windows. Vous pouvez exécuter **asntdiff -f** à partir d'une invite de commande Linux, UNIX ou Windows, à partir de z/OS en tant que travail par lots utilisant JCL ou à partir de z/OS sous l'environnement USS (UNIX System Services).

En plus des instructions SELECT, le fichier en entrée contient les informations de base de données source et cible, les informations de différenciation entre les tables et des paramètres facultatifs qui spécifient des méthodes de traitement des différences. Vous pouvez utiliser un fichier de mots de passe créé par la commande **asnpwd** pour spécifier un ID utilisateur et un mot de passe afin de se connecter aux bases de données source et cible.

Remarque : La commande `asntrep` pour la réparation des différences entre les tables ne prend pas en charge l'option de fichier en entrée.

Le format du contenu du fichier en entrée est le suivant :

```
* Optional comment line
# Optional comment line
SOURCE_SERVER=nom_serveur
SOURCE_SELECT="SQL_SELECT_STATEMENT"
TARGET_SERVER=nom_serveur
TARGET_SELECT="SQL_SELECT_STATEMENT"
PARAMETER=valeur
...
```

Suivez ces instructions :

- Chaque paramètre doit suivre le format *paramètre=valeur*.
- Plusieurs paires paramètre-valeur peuvent être spécifiées sur une seule ligne, en les séparant par un espace vide. Les paires paramètre-valeur peuvent également être spécifiées sur une nouvelle ligne.
- Pour conserver les espaces vides, placez les valeurs de paramètre entre guillemets ("). Les guillemets sont également requis pour les instructions SELECT source et cible.
- Si vous voulez conserver l'utilisation d'une casse mixte ou d'espaces vides dans les noms d'objets DB2 uniques (noms de table ou de colonne, DIFF_SCHEMA, DIFF_TABLESPACE) masquez-les avec \" \", par exemple \"MY NAME\" ou \"ColumnName\" ou \"name\".
- Les commentaires doivent être précédés d'un astérisque (*) ou d'un signe dièse (#). Cette ligne est ignorée. Les commentaires doivent figurer sur leur propre ligne et ne peuvent pas être ajoutés sur une ligne contenant des paramètres.
- Placez les paramètres DIFF_PATH et PWDFILE entre guillemets ("). Un délimiteur de chemin final pour DIFF_PATH n'est pas obligatoire.

Syntaxe

► `asntdiff -f nom_fichier_entrée` ◄

Paramètres

Le tableau 49 définit les paramètres obligatoires à inclure dans le fichier en entrée pour la commande **asntdiff -f**.

Pour des descriptions des paramètres facultatifs que vous pouvez inclure dans le fichier en entrée (et qui sont partagés par la commande `asntdiff` standard), voir «`asntdiff` : comparaison des données dans les tables source et cible (z/OS)», à la page 454 ou «`asntdiff` : comparaison des données dans les tables source et cible (Linux, UNIX, Windows)», à la page 450.

Tableau 49. Définitions des paramètres d'appel `asntdiff -f` pour Linux, UNIX, Windows et z/OS

Paramètre	Définition
<i>nomfichier_entrée</i>	Spécifie le nom du fichier contenant les informations de base de données source et cible et les instructions SELECT. Spécifiez un chemin de répertoire si le fichier se trouve à un autre emplacement que le répertoire à partir duquel vous exécutez la commande asntdiff -f .
SOURCE_SERVER= <i>nom_serveur_source</i>	Spécifie l'alias de la base de données où se trouve la table source.
TARGET_SERVER= <i>nom_serveur_cible</i>	Spécifie l'alias de la base de données où se trouve la table cible.
SOURCE_SELECT= <i>instruction_sélection_source</i> TARGET_SELECT= <i>instruction_sélection_cible</i>	<p>Toute instruction SELECT SQL valide.</p> <p>Les ensembles de résultats provenant de l'instruction SQL sur chaque table doivent contenir des colonnes avec des longueurs et des types de données correspondants. La commande asntdiff décrit les requêtes et compare les données provenant des deux ensembles de résultats. La commande ne vérifie pas de manière explicite le catalogue système en ce qui concerne les informations de longueur et de type. L'instruction SELECT peut être une instruction ouverte comme dans (*) ou une instruction SELECT contenant des noms de colonne, des expressions SQL et des clauses WHERE autorisés.</p> <p>Une clause ORDER BY est obligatoire. La clause doit contenir les valeurs numériques des positions des colonnes dans l'instruction SQL.</p> <p>Assurez-vous que la colonne ou les colonnes de la clause ORDER BY font référence à une clé unique ou à une clé composée unique. Dans le cas contraire, les résultats seraient incorrects. Un index sur les colonnes de la clause ORDER BY peut améliorer les performances en éliminant la nécessité d'un tri.</p> <p>L'ensemble de l'instruction doit être placé entre guillemets pour marquer le début et la fin.</p>

Les exemples suivants illustrent les paramètres obligatoires, les instructions SQL et les paramètres facultatifs que vous placez dans le fichier en entrée.

z/OS

Exemple 1

Cet exemple illustre l'utilisation d'une instruction SELECT ouverte sous DB2 for z/OS. Notez l'utilisation de \" pour conserver la mixité de la casse pour le propriétaire de la table et l'utilisation de paramètres facultatifs dans le fichier en entrée. Notez également l'utilisation du paramètre DB2_SUBSYSTEM.

```
SOURCE_SERVER=STPLEX4A_DSN7
SOURCE_SELECT="select * from CXAIMS.ALDEC order by 1"
TARGET_SERVER=STPLEX4A_DSN7
TARGET_SELECT="select * from \"Cxaims\".TARG_ALDEC order by 1"
DIFF_DROP=Y
DB2_SUBSYSTEM=DSN7
MAXDIFF=10000
DEBUG=YES
```

z/OS

Exemple 2

Cet exemple illustre l'utilisation des fonctions SUBSTR et CAST dans les instructions SELECT.

```
SOURCE_SERVER=D7DP
SOURCE_SELECT="select HIST_CHAR12,HIST_DATE,HIST_CHAR6,HIST_INT1,HIST_INT2,
HIST_INT3,SUBSTR(CHAR1,1,5) AS CHAR1,SUBSTR(CHAR2,1,10) AS CHAR2,HIST_INT3,
HIST_DEC1,HIST_DEC2,HIST_DEC3,CAST(INT1 AS SMALLINT) AS INT1
FROM BISVT.THIST17 ORDER BY 4"
TARGET_SERVER=STPLEX4A_DSN7
TARGET_SELECT="select HIST_CHAR12,HIST_DATE,HIST_CHAR6,HIST_INT1,HIST_INT2,
HIST_INT3,CHAR1,CHAR2,HIST_INT3,HIST_DEC1,HIST_DEC2,HIST_DEC3,SML1
FROM BISVT.THIST17 ORDER BY 4"
DB2_SUBSYSTEM=DSN7
DIFF_DROP=Y
DEBUG=YES
MAXDIFF=10000
```

Windows

Exemple 3

Cet exemple compare les tables EMPLOYEE sur SOURCEDB et TARGETDB et inclut plusieurs paramètres facultatifs.

```
SOURCE_SERVER=SOURCEDB
SOURCE_SELECT="select FIRSTNME, LASTNAME, substr(WORKDEPT,1,1)
as WORKDEPT, EMPNO from EMPLOYEE order by 4"
TARGET_SERVER=TARGETDB
TARGET_SELECT="select FIRSTNME, LASTNAME, substr(WORKDEPT,1,1)
as WORKDEPT, EMPNO from EMPLOYEE order by 4"
DIFF_DROP=Y
DIFF_=\"diffTable\"
DEBUG=YES
MAXDIFF=10000
PWDFILE="asnpwd.aut"
DIFF_PATH="C:\utils\"
```

Linux UNIX

Exemple 4

Cet exemple compare les tables EMPLOYEE dans un environnement Linux ou UNIX et utilise une fonction de transtypage.

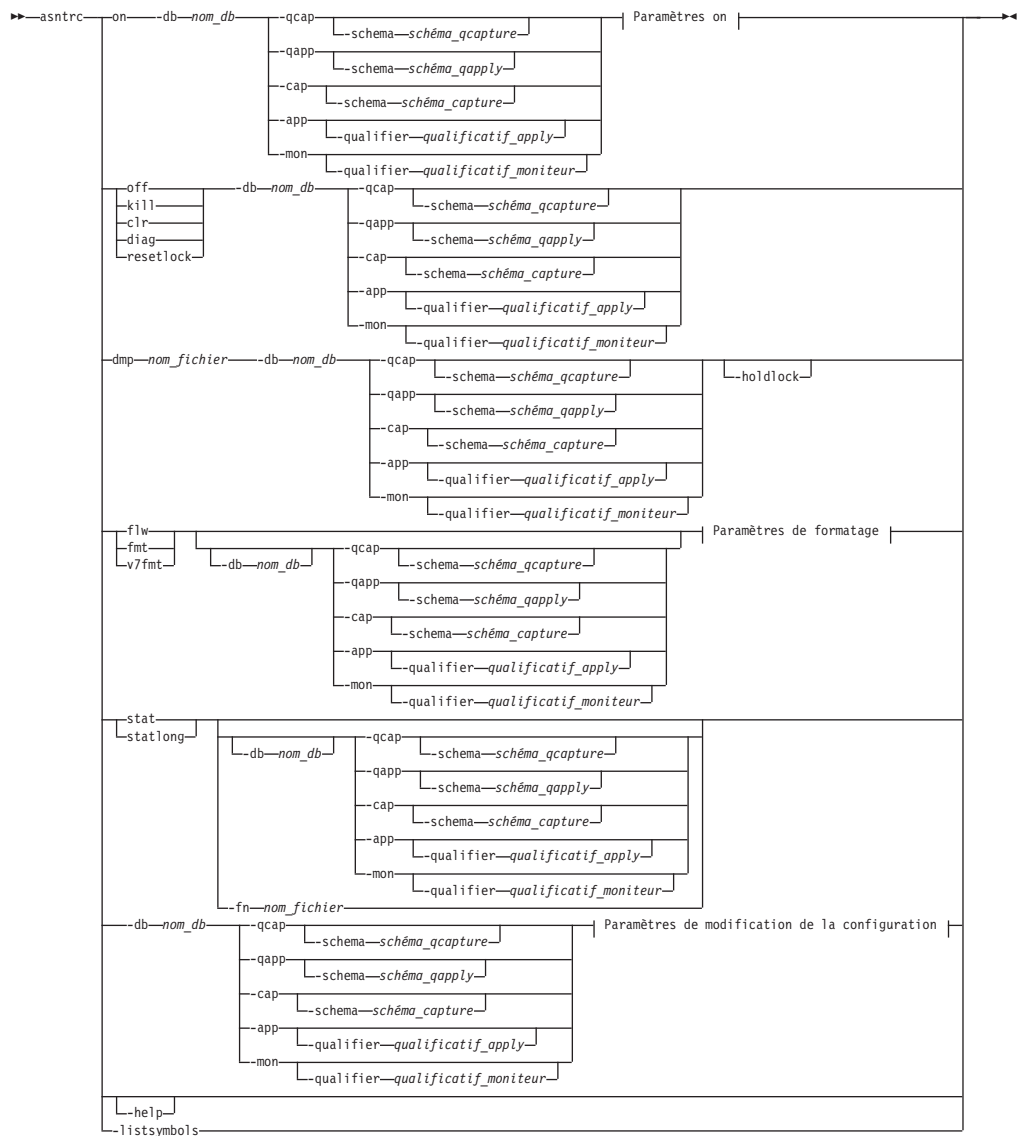
```
SOURCE_SERVER=SOURCEDB
SOURCE_SELECT="select EMPNO, FIRSTNME, LASTNAME, cast(SALARY as INT)
as SALARY from EMPLOYEE order by 1"
TARGET_SERVER=TARGETDB
TARGET_SELECT="select EMPNO, FIRSTNME, LASTNAME, cast(SALARY as INT)
as SALARY from EMPLOYEE order by 1"
DIFF_DROP=Y
DIFF_ =\"diffTable\"
DEBUG=YES
MAXDIFF=10000
PWDFILE="asnpwd.aut"
DIFF_PATH="home/laxmi/utlils"
```

asntrc : Utilisation de la fonction trace de réplication

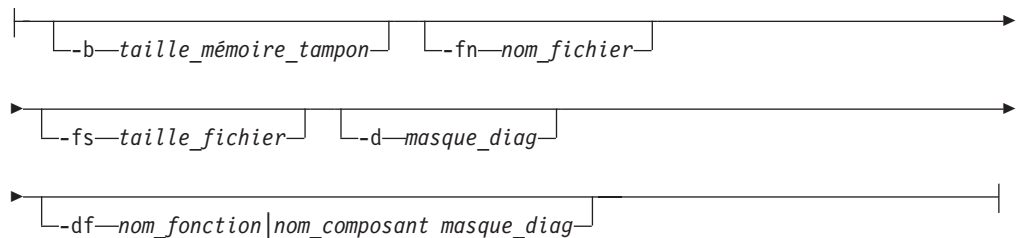
Utilisez la commande **asntrc** pour exécuter l'utilitaire trace sous Linux, UNIX, Windows et UNIX System Services (USS) sous z/OS. La fonction trace consigne les informations sur le flux de programme depuis les programmes Q Capture, Q Apply, Capture, Apply, et Replication Alert Monitor. Vous pouvez fournir ces informations de trace au service de support logiciel IBM pour obtenir une assistance à la résolution des incidents. Exécutez cette commande à l'invite du système d'exploitation ou dans un script de shell.

Cette commande s'exécute à l'invite du système d'exploitation ou dans un script de shell.

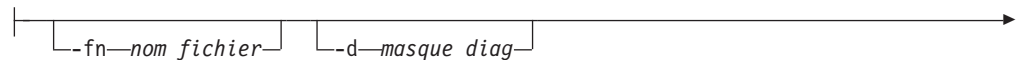
Syntaxe



Paramètres On :



Paramètres de formatage :



```
└─df─nom_fonction|nom_composant masque_diag┘ └─holdlock┘
```

Paramètres de modification de la configuration :

```
└─d─masque_diag┘ └─df─nom_fonction|nom_composant masque_diag┘
```

Paramètres

Le tableau 50 définit les paramètres d'appel de la commande **asntrc**.

Tableau 50. définition des paramètres d'appel de la commande *asntrc* pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS

Paramètre	Définition
on	Indiquez ce paramètre pour activer l'utilitaire trace pour un programme Q Capture, Q Apply, Capture, Apply ou Moniteur d'alertes de réplication spécifique. Cet utilitaire trace crée un segment de mémoire partagé durant le processus de traçage.
-db <i>nom_db</i>	Indique le nom de la base de données à tracer : <ul style="list-style-type: none"> • Indique le nom du serveur Q Capture pour le programme Q Capture à tracer. • Indique le nom du serveur Q Apply pour le programme Q Apply à tracer. • Indique le nom du serveur de contrôle Capture pour le programme Capture à tracer. • Indique le nom du serveur de contrôle Apply pour le programme Apply à tracer. • Indique le nom du serveur de contrôle Moniteur pour le programme Moniteur d'alertes de réplication à tracer.
-qcap	Indique qu'un programme Q Capture doit être tracé. Le programme Q Capture est identifié par le paramètre -schema .
-schema <i>schéma_qcapture</i>	Indique le nom du programme Q Capture à tracer. Le programme Q Capture est identifié par le schéma Q Capture que vous entrez. Utilisez ce paramètre avec le paramètre -qcap .
-qapp	Indique qu'un programme Q Apply doit être tracé. Le programme Q Apply est identifié par le paramètre -schema .
-schema <i>schéma_qapply</i>	Indique le nom du programme Q Apply à tracer. Le programme Q Apply est identifié par le schéma Q Apply que vous entrez. Associez ce paramètre à -qapp .
-cap	Indique qu'un programme Capture doit être tracé. Le programme Capture est identifié par le paramètre -schema .
-schema <i>schéma_capture</i>	Indique le nom du programme Capture à tracer. Le programme Capture est identifié par le schéma Capture que vous entrez. Associez ce paramètre à -cap .

Tableau 50. définition des paramètres d'appel de la commande *asntrc* pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
-app	Indique qu'un programme Apply doit être tracé. Le programme Apply est identifié par le paramètre -qualifier .
-qualifier <i>qualificatif_apply</i>	Indique le nom du programme Apply à tracer. Ce programme Apply est identifié par le qualificatif Apply que vous entrez. Associez ce paramètre à -app .
-mon	Indique qu'un programme Moniteur d'alertes de réplication doit être tracé. Le programme Moniteur d'alertes de réplication est identifié par le paramètre -qualifier .
-qualifier <i>qualificatif_moniteur</i>	Indique le nom du programme Moniteur d'alertes de réplication à tracer. Ce programme est identifié par le qualificatif du moniteur que vous entrez. Associez ce paramètre à -mon .
off	Indiquez ce paramètre pour désactiver l'utilitaire trace pour un programme Q Capture, Q Apply, Capture, Apply ou Moniteur d'alertes de réplication spécifique et libérer le segment de mémoire partagé en cours d'utilisation.
kill	Indiquez ce paramètre pour forcer une interruption anormale de l'utilitaire trace. N'utilisez ce paramètre qu'en cas de problème et que si vous ne parvenez pas à désactiver l'utilitaire trace avec le paramètre off .
clr	Indiquez ce paramètre pour effacer une mémoire tampon. Ce paramètre efface le contenu de la mémoire tampon, mais laisse cette dernière activée.
diag	Indiquez ce paramètre pour visualiser les paramètres du filtre pendant l'exécution de l'utilitaire trace.
resetlock	Indiquez ce paramètre pour dégager le verrou de la mémoire tampon d'un utilitaire trace. Ce paramètre active le verrouillage de la mémoire tampon pour restaurer le système suite à une fermeture prématurée du programme trace tout en tenant le verrou de la mémoire tampon.
dmp <i>nom_fichier</i>	Indiquez ce paramètre pour écrire le contenu actuel de la mémoire tampon de trace dans un fichier.
-holdlock	Indique que l'utilitaire trace peut vider un fichier ou exécuter une commande de sortie tout en tenant un verrou, même s'il ne dispose pas d'une mémoire suffisante pour copier la mémoire tampon.
flw	Indiquez ce paramètre pour afficher les informations récapitulatives générées par l'utilitaire trace et conservées dans la mémoire partagée ou un fichier. Ces informations incluent le flux du programme, et s'affichent avec des mises en retrait qui montrent la fonction et les structures de piles d'appels pour chaque processus et unité d'exécution.

Tableau 50. définition des paramètres d'appel de la commande *asnrtrc* pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
fmt	Indiquez ce paramètre pour afficher les informations détaillées générées par l'utilitaire trace et conservées dans la mémoire partagée ou un fichier. Ce paramètre affiche le contenu entier des structures de données tracées par ordre chronologique.
v7fmt	Indiquez ce paramètre pour afficher les informations générées par l'utilitaire trace et enregistrées dans la mémoire partagée ou dans un fichier. Ces informations de trace apparaissent au format de la version 7.
stat	Indiquez ce paramètre pour afficher l'état d'un utilitaire trace. Ces informations d'état incluent la version de trace, la version de l'application, le nombre d'entrées, la taille de la mémoire tampon, la quantité de mémoire tampon utilisée, le code d'état et l'horodatage du programme.
statlong	Indiquez ce paramètre pour afficher l'état d'un utilitaire trace avec des informations supplémentaires sur le niveau de version z/OS. Ces informations supplémentaires incluent les niveaux de service de chaque module de l'application et s'affichent sous forme de chaînes de texte longues.
-fn <i>nom_fichier</i>	Indique le nom de fichier contenant les informations de trace mises en miroir, qui incluent l'ensemble des informations de sortie de l'utilitaire trace.
-help	Affiche les paramètres de la commande valide avec des descriptions.
-listsymbols	Affiche la fonction et les identificateurs de composants valides à utiliser avec le paramètre -df .
-b <i>taille_mémoire_tampon</i>	Indique la taille de la mémoire tampon de trace (en octets). Vous pouvez saisir la lettre K ou M après le numéro pour indiquer les kilooctets ou mégaoctets, respectivement. Ces lettres ne sont pas sensibles à la casse.
-fs <i>taille_fichier</i>	Indique la taille limite (en octets) du fichier d'informations de trace en miroir.

Tableau 50. définition des paramètres d'appel de la commande `asntrc` pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<code>-d masque_diag</code>	<p>Indique les types d'enregistrements de trace qui doivent être consignés par l'utilitaire trace. Les enregistrements de trace sont classés suivant un numéro de masque de diagnostic :</p> <p>1 Données de flux, qui comprennent les points d'entrée et de sortie des fonctions.</p> <p>2 Données de base, qui incluent tous les événements majeurs rencontrés par l'utilitaire trace.</p> <p>3 Données détaillées, qui comprennent les événements majeurs et leurs descriptions.</p> <p>4 Données de performance.</p> <p>Important : Les numéros de masques de diagnostic plus élevés n'incluent <i>pas</i> les numéros de masques de diagnostic inférieurs.</p> <p>Vous pouvez entrer un ou plusieurs numéros pour créer un masque de diagnostic qui inclue uniquement les enregistrements de trace nécessaires. Par exemple, saisissez <code>-d 4</code> pour n'enregistrer que les données de performance, <code>-d 1,4</code> pour n'enregistrer que les données de flux et de performance ou <code>-d 1,2,3,4</code> (valeur par défaut) pour enregistrer tous les enregistrements de trace. Séparez les numéros par des virgules.</p> <p>Entrez le numéro de masque de diagnostic 0 (zéro) pour indiquer qu'aucun enregistrement de trace global ne doit être enregistré par l'utilitaire trace. Tapez <code>-d 0</code> pour réinitialiser le niveau de diagnostic avant d'indiquer de nouveaux numéros de masques de diagnostic pour un utilitaire de traçage.</p>
<code>-df nom_fonction nom_composant masque_diag</code>	<p>Indique qu'un identificateur de fonction ou de composant particulier doit être tracé.</p> <p>Tapez le numéro de masque de diagnostic (1,2,3,4) après le nom de l'identificateur de fonction ou de composant. Vous pouvez entrer un ou plusieurs numéros. Séparez les numéros par des virgules.</p>

Exemples de commande `asntrc`

Les exemples suivants illustrent comment utiliser la commande `asntrc`. Ces exemples peuvent être exécutés sur les systèmes d'exploitation Linux, UNIX, Windows ou z/OS.

Exemple 1

Pour tracer un programme Capture en cours d'exécution :

1. Lancez l'utilitaire trace en indiquant un nom de fichier trace, ainsi qu'une taille de fichier et de mémoire tampon maximale :

```
asntrc on -db mydb -cap -schema myschema -b 256k -fn myfile.trc -fs 500m
```

2. Lancez le programme Capture et laissez-le activé pendant une période appropriée.
3. Pendant l'exécution de l'utilitaire trace, affichez les données directement à partir de la mémoire partagée.

Pour afficher le récapitulatif du processus et des informations sur l'unité d'exécution à partir de l'utilitaire trace :

```
asnlrc flw -db mydb -cap -schema myschema
```

Pour visualiser les enregistrements de données de flux, et de performance de base et détaillées uniquement à partir du lecteur de journal Capture :

```
asnlrc fmt -db mydb -cap -schema myschema -d 0
-df "Capture Log Read" 1,2,3,4
```

4. Arrêter l'utilitaire trace :

```
asnlrc off -db mydb -cap -schema myschema
```

Le fichier trace contient toutes les données de trace du programme Capture qui ont été générées depuis le démarrage du programme Capture jusqu'à la désactivation de l'utilitaire trace.

5. Après l'arrêt de l'utilitaire trace, formatez les données du fichier binaire généré :

```
asnlrc flw -fn myfile.trc
```

et

```
asnlrc fmt -fn myfile.trc -d 0 -df "Capture Log Read" 1,2,3,4
```

Exemple 2

Pour démarrer un utilitaire trace d'un programme Moniteur d'alertes de réplication :

```
asnlrc on -db mydb -mon -qualifier monq
```

Exemple 3

Pour tracer uniquement les données de performance d'un programme Apply :

```
asnlrc on -db mydb -app -qualifier aql -b 256k -fn myfile.trc -d 4
```

Exemple 4

Pour tracer toutes les données de flux et de performance d'un programme Capture :

```
asnlrc on dbserve1 -cap -schema myschema -b 256k
-fn myfile.trc -d 1,4
```

Exemple 5

Pour tracer toutes les données de performance globales et les données de flux du lecteur de journal Capture propres à un programme Capture :

```
asnlrc on -db mydb -cap -schema myschema -b 256k -fn myfile.trc -d 4
-df "Capture Log Read" 1
```

Exemple 6

Pour tracer un programme Capture en cours d'exécution, puis afficher et enregistrer une image instantanée de l'utilitaire trace :

1. Lancez la commande trace en indiquant une taille de mémoire tampon suffisante pour contenir les enregistrements les plus récents :
`asntrc on -db mydb -cap -schema myschema -b 4m`
2. Lancez le programme Capture et laissez-le activé pendant une période appropriée.
3. Consultez les informations de trace instantanées détaillées qui sont stockées dans la mémoire partagée :
`asntrc fmt -db mydb -cap -schema myschema`
4. Enregistrez les informations de trace instantanées dans un fichier :
`asntrc dmp myfile.trc -db mydb -cap -schema myschema`
5. Arrêter l'utilitaire trace :
`asntrc off -db mydb -cap -schema myschema`

Exemples pour `asntrc` avec des segments partagés

L'utilitaire trace autonome **asntrc** utilise un segment partagé pour communiquer avec les programmes respectifs Q Capture, Q Apply, Capture, Apply ou Moniteur d'alertes de réplication à tracer. Le segment partagé sera également utilisé pour contenir les entrées de trace si aucun fichier n'est spécifié. Autrement, vous devez indiquer des options correspondantes pour la commande **asntrc** et les programmes respectifs à tracer pour faire correspondre le segment partagé adéquat aux traces de contrôle. Les exemples suivants montrent les options qui doivent être spécifiées lorsque l'utilitaire trace est combiné avec les programmes Q Capture, Q Apply, Capture, Apply ou Moniteur d'alertes de réplication.

Pour le programme Q Capture, la base de données spécifiée par le paramètre **-db** avec la commande **asntrc** doit correspondre à celle qui est spécifiée par le paramètre **capture_server** avec la commande **asnqcap** :

```
asntrc -db ASN6 -schema EMI -qcap
asnqcap capture_server=ASN6 capture_schema=EMI
```

Pour le programme Q Apply, la base de données spécifiée par le paramètre **-db** avec la commande **asntrc** doit correspondre à celle qui est spécifiée par le paramètre **apply_server** avec la commande **asnqapp** :

```
asntrc -db TSN3 -schema ELB -qapp
asnqapp apply_server=TSN3 apply_schema=ELB
```

Pour le programme Capture, la base de données spécifiée par le paramètre **-db** avec la commande **asntrc** doit correspondre à celle qui est spécifiée par le paramètre **capture_server** avec la commande **asnqcap** :

```
asntrc -db DSN6 -schema JAY -cap
asnqcap capture_server=DSN6 capture_schema=JAY
```

Pour le programme Apply, la base de données spécifiée par le paramètre **-db** avec la commande **asntrc** doit correspondre à celle qui est spécifiée par le paramètre **control_server** avec la commande **asnqapp** :

```
asntrc -db SVL_LAB_DSN6 -qualifier MYQUAL -app
asnqapp control_server=SVL_LAB_DSN6 apply_qual=MYQUAL
```

Pour le programme Moniteur d'alertes de réplication, la base de données spécifiée par le paramètre **-db** avec la commande **asntrc** doit correspondre à celle qui est spécifiée par le paramètre **monitor_server** avec la commande **asnmon** :

```
asntrc -db DSN6 -qualifier MONQUAL -mon
asnmon monitor_server=DSN6 monitor_qual=MONQUAL
```

asntrep : réparation des différences entre les tables source et cible

La commande **asntrep** permet de synchroniser une table source et une table cible en corrigeant les différences existantes entre les deux. Exécutez la commande **asntrep** sous Linux, UNIX et Windows à l'invite du système d'exploitation ou dans un script de shell.

Syntaxe

```

▶ asntrep—DB=serveur—DB2_SUBSYSTEM=sous-système—SCHEMA=schéma—
DIFF_SCHEMA=schéma_table_différences—DIFF_TABLESPACE=espace_table—
WHERE=clause_WHERE—DIFF_PATH=chemin_journal—PWDFILE=nom_fichier—
DIFF=nom_table—

```

Paramètres

Le tableau 51 définit les paramètres d'appel de la commande **asntrep**.

Tableau 51. Définitions des paramètres d'appel de la commande **asntrep** pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Définition
DB=serveur	Indique l'alias DB2 de la base de données contenant des informations sur les tables source et cible que vous souhaitez synchroniser. La valeur diffère selon que vous utilisez la réplification Q ou la réplification SQL : Réplification Q La valeur est le nom du serveur Q Capture qui contient la table IBMQREP_SUBS. Réplification SQL La valeur est le nom du serveur de contrôle Apply qui contient la table IBMSNAP_SUBS_MEMBR. z/OS La valeur de ce paramètre est un nom d'emplacement.
DB2_SUBSYSTEM=sous-système	z/OS Indique le nom du sous-système dans lequel vous exécutez l'utilitaire asntrep.
SCHEMA=schéma	Indique le schéma des tables de contrôle Q Capture pour la réplification Q ou les tables de contrôle Apply pour la réplification SQL.
DIFF_SCHEMA=schéma_table_différences	Indique le schéma qui qualifie la table des différences. Il s'agit par défaut de ASN.

Tableau 51. Définitions des paramètres d'appel de la commande *asntrep* pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Définition
DIFF_TABLESPACE = <i>espace_table</i>	Indique l'espace table dans lequel une copie de la table des différences est placée dans la base de données ou le sous-système cible. La copie est ensuite utilisée pour réparer la table cible. Si ce paramètre n'est pas spécifié, la table est créée dans l'espace table par défaut de la base de données ou du sous-système dans lequel la commande asntrep a été exécutée.
WHERE = <i>clause_WHERE</i>	<p>Indique une clause WHERE SQL qui identifie de manière unique une ligne de la table de contrôle contenant des informations sur les tables source et cible que vous synchronisez. La clause WHERE doit être entourée de guillemets doubles. La valeur de ce paramètre diffère selon que vous utilisez la réplication Q ou la réplication SQL :</p> <p>Réplication Q La clause WHERE indique une ligne de la table IBMQREP_SUBS en utilisant la colonne SUBNAME pour identifier l'abonnement Q qui contient les tables source et cible.</p> <p>Réplication SQL La clause WHERE indique une ligne de la table IBMSNAP_SUBS_MEMBR en utilisant les colonnes SET_NAME, APPLY_QUAL, TARGET_SCHEMA et TARGET_TABLE pour identifier le membre de l'ensemble d'abonnements qui contient les tables source et cible.</p>
DIFF_PATH = <i>chemin_journal</i>	Indique l'emplacement auquel vous souhaitez que l'utilitaire <i>asntrep</i> écrive son journal. La valeur par défaut est le répertoire dans lequel vous avez exécuté la commande. La valeur doit être un nom de chemin absolu. Pour conserver la casse, utilisez des guillemets doubles ("").
PWDFILE = <i>nom_fichier</i>	Indique le nom du fichier de mot de passe qui permet de se connecter aux bases de données. Si vous ne spécifiez aucun fichier de mot de passe, la valeur par défaut est <i>asnpwd.aut</i> (nom du fichier de mot de passe qui est créé par la commande asnpwd). L'utilitaire <i>asntrep</i> recherche le mot de passe dans le répertoire qui est spécifié par le paramètre DIFF_PATH . Si aucune valeur n'est spécifiée pour le paramètre DIFF_PATH , la commande recherche le fichier de mot de passe dans le répertoire où la commande a été exécutée.
DIFF = <i>nom_table</i>	Indique le nom de la table qui a été créée dans la base de données source en utilisant la commande asntdiff pour enregistrer les différences entre les tables source et cible. Les informations qui sont mémorisées dans cette table serviront à synchroniser les tables source et cible.

Exemples de commande asntrep

Les exemples suivants illustrent comment utiliser la commande **asntrep**.

Exemple 1

Dans la réplication Q, pour synchroniser une table source et une table cible qui sont spécifiées dans un abonnement Q nommé `my_qsub`, sur un serveur Q Capture nommé `source_db`, avec le schéma Q Capture `asn` et dont les différences sont enregistrées dans une table appelée `q_diff_table` :

```
asntrep db=source_db schema=asn where="subname = 'my_qsub'" diff=q_diff_table
```

Exemple 2

Dans la réplication SQL, pour synchroniser une table source et une table cible qui sont spécifiées dans un ensemble d'abonnements appelé `my_set`, avec une table cible nommée `trg_table`, sur un serveur de contrôle Apply appelé `apply_db`, avec le schéma Apply `asn` et dont les différences sont enregistrées dans une table appelée `sql_diff_table` :

```
asntrep DB=apply_db SCHEMA=asn WHERE="set_name = 'my_set'  
and target_table = 'trg_table'" diff=sql_diff_table
```

asnqanalyze : fonctionnement de l'analyseur de réplication Q (Replication Analyzer)

Utilisez la commande **asnqanalyze** pour réunir des informations sur l'état d'un environnement de réplication Q ou de publication d'événements. Vous pouvez aussi utiliser la commande pour générer un rapport HTML mis en forme sur les tables de contrôle Q Capture ou Q Apply, les catalogues DB2, les fichiers journaux de diagnostic des programmes de réplication et les gestionnaires de file d'attente WebSphere MQ.

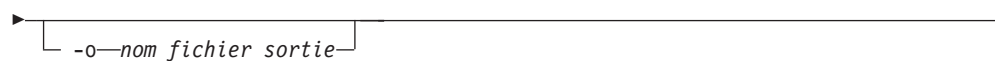
La commande est exécutée sur les systèmes d'exploitation Linux, UNIX ou Windows. Toutefois, il peut se connecter à un sous-système DB2 répertorié localement sous z/OS pour effectuer l'analyse des tables de contrôle.

La commande **asnqanalyze** se compose de trois parties :

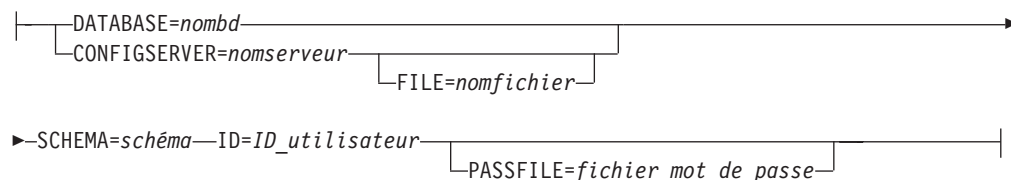
- L'option **GATHER**, chargée de l'analyse et du stockage des résultats dans un ou plusieurs fichiers XML figurant dans le répertoire dans lequel la commande a été exécutée.
- L'option **REPORT**, qui génère un rapport HTML à partir de la sortie de l'option **GATHER**.
- L'option **GENERATE HTML REPORT**, qui génère un rapport HTML des données des tables de contrôle Q Capture et Q Apply sans utiliser l'option **GATHER**.

Utilisez l'option **GATHER** pour chaque schéma inclus dans la configuration de réplication Q. Puis, spécifiez les fichiers de sortie pour chaque appel de l'option **GATHER** dans l'option **REPORT**. Vous pouvez utiliser l'option **GENERATE HTML REPORT** pour générer un rapport HTML brut au lieu d'un rapport détaillé généré par les options **GATHER** plus **REPORT**.

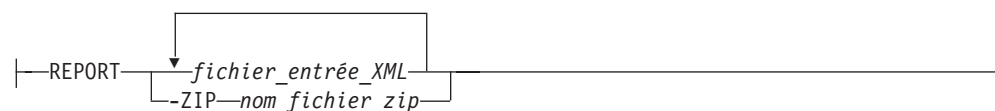
Exécutez la commande **asnqanalyze** à l'invite d'un système d'exploitation.



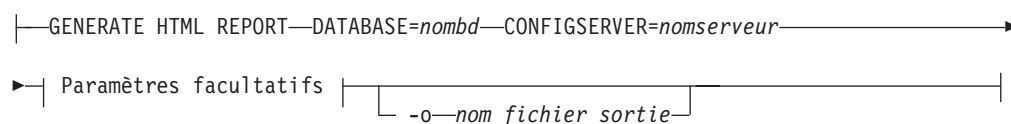
database-options :



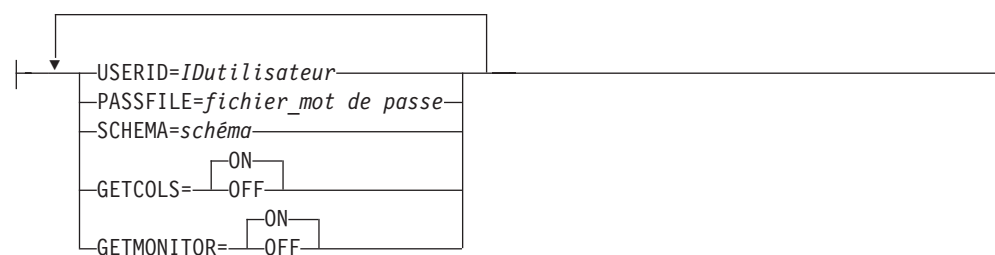
report-options :



generate-html-report-options :



Paramètres facultatifs



Paramètres de la commande gather, base de données unique

Le tableau 52 définit les paramètres d'appel pour la commande `asnqanalyze gather` pour `single_database_options`. Linux UNIX Windows

Tableau 52. Définitions des paramètres d'appel `asnqanalyze gather` pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Description
<code>DATABASE=nombd</code>	Serveur Q Capture ou Q Apply dont les tables de contrôle, les catalogues de base de données ou les fichiers journaux sont analysés.

Tableau 52. Définitions des paramètres d'appel *asnqanalyze gather* pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Description
CONFIGSERVER = <i>nomserveur</i>	Serveur Q Capture ou Q Apply à analyser lorsque asnqanalyze se connecte au serveur via une connexion Type 4.
FILE = <i>nom_fichier</i>	Fichier de configuration dans lequel se trouvent le port, le nom d'hôte et d'autres informations relatives à la connexion. Obligatoire pour les bases de données utilisant une connexion Type 4.
INSTANCE = <i>instance_db2</i>	Linux UNIX Windows L'instance DB2 de la base de données à analyser. Valeur par défaut : DB2
SUBSYSTEM = <i>sous-système_db2</i>	z/OS Le sous-système DB2 à analyser. Valeur par défaut : DSN1
USERID = <i>IDutilisateur</i>	ID utilisateur de connexion à la base de données.
PASSFILE = <i>fichier_mot de passe</i>	Nom du fichier contenant le mot de passe associé à l'ID utilisateur. Pour créer un fichier de mot de passe, créez un fichier ASCII contenant uniquement le mot de passe. Enregistrez ce fichier dans le répertoire dans lequel vous exécutez la commande gather . Le mot de passe ne peut pas être suivi de blancs de fin, de caractères spéciaux ou de retours chariot. Q Analyzer ne peut pas utiliser les fichiers des mots de passe codés créés par la commande asnpwd . Vous devez spécifier un chemin d'accès au fichier des mots de passe lorsque vous exécutez la commande asnqanalyze à partir d'un autre répertoire que celui dans lequel se trouve le fichier. Conseil : L'utilisation d'un fichier des mots de passe se justifie peut-être par le fait que les commandes exécutées à partir de l'invite sont enregistrées dans l'historique, et que les mots de passe sont conservés. Un fichier des mots de passe évite également de saisir le mot de passe chaque fois que vous utilisez Q Analyzer.
SCHEMA = <i>schéma</i>	Schéma des tables de contrôle Q Capture ou Q Apply à analyser. Valeur par défaut : ASN
CAPLOGDIR = <i>répertoire_journal_capture</i>	Répertoire dans lequel le programme Q Capture enregistre son fichier journal de diagnostic. Si Q Capture ne fonctionne pas sur le même système que celui sur lequel vous émettez la commande asnqanalyze , vous devez envoyer le fichier journal à l'un des répertoires du système sur lequel fonctionne Q Capture ; ce paramètre spécifie ce répertoire. Si vous envoyez le fichier journal via FTP, utilisez le mode ASCII. Si aucun fichier journal n'est transféré, les données de consignation sont recherchées dans la table IBMQREP_CAPTRACE. Les données de cette table ne sont pas aussi détaillées que celles contenues dans les fichiers journaux de diagnostic.

Tableau 52. Définitions des paramètres d'appel *asnqanalyze gather* pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Description
APPLGDIR= <i>répertoire_journal_apply</i>	Répertoire dans lequel le programme Q Apply enregistre son fichier journal de diagnostic. Si Q Apply ne fonctionne pas sur le même système que celui sur lequel vous émettez la commande asnqanalyze , vous devez envoyer le fichier journal à l'un des répertoires du système sur lequel fonctionne Q Apply ; ce paramètre spécifie ce répertoire. Si vous envoyez le fichier journal via FTP, utilisez le mode ASCII. Si aucun fichier journal n'est transféré, les données de consignation sont recherchées dans la table IBMQREP_APPLYTRACE. Les données de cette table ne sont pas aussi détaillées que celles contenues dans les fichiers journaux de diagnostic.
PORT=port	Numéro du port sur lequel le programme d'écoute du gestionnaire de files d'attente WebSphere MQ effectue l'écoute des messages entrants. Vous ne devez spécifier un port que si le serveur à analyser est éloigné de l'ordinateur sur lequel vous exécutez la commande asnqanalyze . Valeur par défaut : 1414 Avertissement : Le gestionnaire de files d'attente doit être actif pour que la commande gather puisse l'analyser.
CHANNEL=canal	Canal de connexion du serveur distant utilisé par le gestionnaire de files d'attente. Vous ne devez spécifier un canal que si le serveur à analyser est éloigné de l'ordinateur sur lequel vous exécutez la commande asnqanalyze . Valeur par défaut : SYSTEM.DEF.SVRCONN
HOSTNAME=nom d'hôte	Nom d'hôte ou adresse IP du système distant sur lequel réside le serveur à analyser. Ce paramètre est obligatoire pour la connexion à un gestionnaire de files d'attente WebSphere MQ.
WARNERRSONLY=ON/OFF	Type d'enregistrements à extraire du fichier journal de diagnostic. OFF (valeur par défaut) Tous les enregistrements sont extraits, y compris les messages d'erreur, d'avertissement et d'informations, ainsi que les descriptions des actions du programme. ON Seuls les messages d'erreur et d'avertissement sont extraits.
GETCOLS=ON/OFF	Définit si le programme Q Analyzer doit collecter les informations relatives à toutes les colonnes, ou uniquement celles relatives aux colonnes répliquées. ON (valeur par défaut) Les informations relatives à toutes les colonnes des tables source et cible sont collectées, que ces colonnes soient répliquées ou non. OFF Les informations relatives aux colonnes répliquées sont collectées.

Tableau 52. Définitions des paramètres d'appel `asnqanalyze gather` pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Description
<code>GETMONITOR=ON/OFF</code>	Définit si les données des tables <code>IBMQREP_CAPMON</code> , <code>IBMQREP_CAPQMON</code> et <code>IBMQREP_APPLYMON</code> doivent être extraites. ON (valeur par défaut) Les données sont extraites. OFF Les données ne sont pas extraites.
<code>LOGDAYS=entier</code>	Indique le nombre de jours d'enregistrements à extraire des fichiers journaux et des tables <code>IBMQREP_CAPTRACE</code> , <code>IBMQREP_APPLYTRACE</code> , <code>IBMQREP_SIGNAL</code> , <code>IBMQREP_CAPMON</code> , <code>IBMQREP_CAPQMON</code> et <code>IBMQREP_APPLYMON</code> . Valeur par défaut : 3
<code>ZIP=ON/OFF</code>	Indique si les fichiers XML générés doivent être compressés dans un fichier zip unique (par défaut). Si vous spécifiez <code>ZIP=OFF</code> , la commande génère des fichiers XML individuels.
<code>-o nom_fichier_sortie</code>	Indique le nom du fichier généré par l'option <code>gather</code> . Si vous ne spécifiez aucun nom de fichier, le nom par défaut est le suivant : Fichier compressé <code>nom_base de données.schema.zip</code> Fichier non compressé <code>nom_base de données.schema.xml</code>

Paramètres de la commande `gather`, plusieurs bases de données

Le tableau 53 définit les paramètres d'appel pour la commande `asnqanalyze gather` pour `multiple_database_options`. Utilisez le paramètre `LIST=(options_base_données)` pour indiquer que plusieurs bases de données sont fournies. Linux UNIX Windows

Tableau 53. Définitions des paramètres d'appel `asnqanalyze gather` pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Description
<code>LIST=(options_base_données)</code>	Linux UNIX Windows Indique que plusieurs bases de données sont fournies. Les options de base de données (<code>options_base_données</code>) doivent être définies entre des parenthèses simples, après le paramètre.
<code>DATABASE=nomdb</code>	Serveur Q Capture ou Q Apply dont les tables de contrôle, les catalogues de base de données ou les fichiers journaux sont analysés.
<code>CONFIGSERVER=nomserveur</code>	Serveur Q Capture ou Q Apply dont les tables de contrôle, les catalogues de base de données ou les fichiers journaux sont analysés avec l'option Generate HTML Report. Vous devez définir soit le paramètre <code>DATABASE</code> , soit le paramètre <code>CONFIGSERVER</code> . Si la commande <code>asnqanalyze</code> est utilisée pour une base de données utilisant une connexion Type 4, le paramètre <code>CONFIGSERVER</code> est utilisé.
<code>SCHEMA=schéma</code>	Schéma des tables de contrôle Q Capture ou Q Apply à analyser. Valeur par défaut : ASN

Tableau 53. Définitions des paramètres d'appel `asnqanalyze gather` pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Description
USERID = <i>IDutilisateur</i>	ID utilisateur de connexion à la base de données.
INIFILE = <i>nom_fichier</i>	Fichier de configuration dans lequel se trouvent le port et le nom d'hôte. Obligatoire pour les bases de données utilisant une connexion Type 4.
PASSFILE = <i>fichier_mot de passe</i>	Nom du fichier contenant le mot de passe associé à l'ID utilisateur. Pour créer un fichier de mot de passe, créez un fichier ASCII contenant uniquement le mot de passe. Enregistrez ce fichier dans le répertoire dans lequel vous exécutez la commande gather . Le mot de passe ne peut pas être suivi de blancs de fin, de caractères spéciaux ou de retours chariot. Q Analyzer ne peut pas utiliser les fichiers des mots de passe codés créés par la commande asnpwd . Vous devez spécifier un chemin d'accès au fichier des mots de passe lorsque vous exécutez la commande asnqanalyze à partir d'un autre répertoire que celui dans lequel se trouve le fichier. Conseil : L'utilisation d'un fichier des mots de passe se justifie peut-être par le fait que les commandes exécutées à partir de l'invite sont enregistrées dans l'historique, et que les mots de passe sont conservés. Un fichier des mots de passe évite également de saisir le mot de passe chaque fois que vous utilisez Q Analyzer.
GETCOLS =ON/OFF	Définit si le programme Q Analyzer doit collecter les informations relatives à toutes les colonnes, ou uniquement celles relatives aux colonnes répliquées. ON (valeur par défaut) Les informations relatives à toutes les colonnes des tables source et cible sont collectées, que ces colonnes soient répliquées ou non. OFF Les informations relatives aux colonnes répliquées sont collectées.
GETMONITOR =ON/OFF	Définit si les données des tables IBMQREP_CAPMON, IBMQREP_CAPQMON et IBMQREP_APPLYMON doivent être extraites. ON (valeur par défaut) Les données sont extraites. OFF Les données ne sont pas extraites.
ZIP =ON/OFF	Indique si les fichiers XML générés doivent être compressés dans un fichier zip unique (par défaut). Si vous spécifiez ZIP=OFF , la commande génère des fichiers XML individuels.
-o <i>nom_fichier_sortie</i>	Indique le nom du fichier généré par l'exécution de la commande asnqanalyze gather . Si vous ne spécifiez aucun nom de fichier, le nom par défaut est le suivant : Fichier compressé <i>nom_base de données.schema.zip</i> Fichier non compressé <i>nom_base de données.schema.xml</i>

Paramètres de la commande report

Le tableau 54 définit les paramètres d'appel de la commande **asnqanalyze report**.

Linux UNIX Windows

Tableau 54. Définitions des paramètres d'appel *asnqanalyze report* pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Description
<i>fichier_entrée_XML</i>	Spécifie un ou plusieurs fichiers XML séparés par un espace ayant été générés par la commande asnqanalyze gather . Vous devez spécifier un chemin d'accès au fichier XML lorsque vous exécutez la commande asnqanalyze report à partir d'un répertoire différent de celui dans lequel se trouve le fichier. L'invite de commande vous demande d'indiquer l'alias du serveur sur lequel le fichier a été créé.
-ZIP <i>fichier_zip</i>	Indique qu'un fichier zippé, généré par la commande asnqanalyze gather , est fourni en entrée. Ce fichier zip peut contenir un ou plusieurs fichiers XML. La fonction REPORT dézippe le fichier et analyse chaque fichier XML.

Paramètres de la commande generate html report

Le tableau 55 définit les paramètres d'appel pour la commande **asnqanalyze generate html report**. Linux UNIX Windows

Tableau 55. Définitions des paramètres d'appel *asnqanalyze gather* pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Description
DATABASE = <i>nombd</i>	Serveur Q Capture ou Q Apply dont les tables de contrôle, les catalogues de base de données ou les fichiers journaux sont analysés.
CONFIGSERVER = <i>nomserveur</i>	Serveur Q Capture ou Q Apply à analyser lorsque asnqanalyze se connecte au serveur via une connexion Type 4.
INIFILE = <i>nom_fichier</i>	Fichier de configuration dans lequel se trouvent le port et le nom d'hôte. Obligatoire pour les bases de données utilisant une connexion Type 4.
USERID = <i>IDutilisateur</i>	ID utilisateur de connexion à la base de données.
PASSFILE = <i>fichier_mot de passe</i>	Nom du fichier contenant le mot de passe associé à l'ID utilisateur. Pour créer un fichier de mot de passe, créez un fichier ASCII contenant uniquement le mot de passe. Enregistrez ce fichier dans le répertoire dans lequel vous exécutez la commande gather . Le mot de passe ne peut pas être suivi de blancs de fin, de caractères spéciaux ou de retours chariot. Q Analyzer ne peut pas utiliser les fichiers des mots de passe codés créés par la commande asnpwd . Vous devez spécifier un chemin d'accès au fichier des mots de passe lorsque vous exécutez la commande asnqanalyze à partir d'un autre répertoire que celui dans lequel se trouve le fichier. Conseil : L'utilisation d'un fichier des mots de passe se justifie peut-être par le fait que les commandes exécutées à partir de l'invite sont enregistrées dans l'historique, et que les mots de passe sont conservés. Un fichier des mots de passe évite également de saisir le mot de passe chaque fois que vous utilisez Q Analyzer.

Tableau 55. Définitions des paramètres d'appel `asnqanalyze gather` pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Description
<code>SCHEMA=schéma</code>	Schéma des tables de contrôle Q Capture ou Q Apply à analyser. Valeur par défaut : ASN
<code>GETCOLS=ON/OFF</code>	Définit si le programme Q Analyzer doit collecter les informations relatives à toutes les colonnes, ou uniquement celles relatives aux colonnes répliquées. ON (valeur par défaut) Les informations relatives à toutes les colonnes des tables source et cible sont collectées, que ces colonnes soient répliquées ou non. OFF Les informations relatives aux colonnes répliquées sont collectées.
<code>GETMONITOR=ON/OFF</code>	Définit si les données des tables IBMQREP_CAPMON, IBMQREP_CAPQMON et IBMQREP_APPLYMON doivent être extraites. ON (valeur par défaut) Les données sont extraites. OFF Les données ne sont pas extraites.
<code>-o nom_fichier_sortie</code>	Indique le nom du fichier généré par l'exécution de la commande <code>asnqanalyze gather</code> . Si vous ne spécifiez aucun nom de fichier, le nom par défaut est le suivant : Fichier compressé <i>nom_base de données.schéma.zip</i> Fichier non compressé <i>nom_base de données.schéma.xml</i>

Exemple 1

La commande suivante analyse les tables de contrôle Q Capture, les catalogues DB2, ainsi que le gestionnaire de files d'attente du serveur Q Capture SAMPLE, pour l'ID utilisateur db2admin et le fichier de mot de passe `pwdfile.txt`, et recueille uniquement les informations relatives aux colonnes répliquées.

Dans cet exemple, la commande est exécutée à partir du répertoire `SQLLIB\bin`, dans lequel le programme Q Capture conserve par défaut son fichier journal ; il est donc inutile de spécifier le paramètre `CAPLOGDIR`. Le fichier de mot de passe est également enregistré dans ce répertoire ; vous n'avez donc pas à spécifier de chemin d'accès à ce fichier.

La commande étant exécutée à partir du même système que celui du gestionnaire de files d'attente, les paramètres `PORT`, `CHANNEL` et `HOSTNAME` ne sont pas non plus spécifiés.

```
asnqanalyze gather DATABASE=SAMPLE USERID="db2admin"
PASSFILE="pwdfile.txt" GETCOLS=OFF
```

Exemple 2

La commande suivante analyse les tables de contrôle Q Apply, les catalogues DB2, ainsi que le gestionnaire de files d'attente du serveur Q Apply TARGET, sous-système distant DSN4 DB2 pour z/OS répertorié localement sous la forme

DSN4_TGT. Le fichier journal de diagnostic Q Apply a été envoyé par FTP au même répertoire que celui à partir duquel la commande est exécutée et dans lequel le fichier de mot de passe est conservé.

Cette commande spécifie également que six jours de données doivent être recueillis à partir des tables IBMQREP_APPLYTRACE et IBMQREP_APPLYMON et que la sortie doit se présenter sous forme de fichier XML non compressé : target.zip.

```
asnqanalyze gather DATABASE=TARGET SUBSYSTEM=DSN4
USERID=db2admin PASSFILE=pwdfilere.txt HOSTNAME="Z145HG"
LOGDAYS=6 ZIP=OFF -o target.zip
```

Exemple 3

Pour générer un rapport HTML qui utilise la sortie de la commande **gather** de l'exemple 1 :

```
asnqanalyze report SAMPLE.ASN.xml
```

Exemple 4

Les exemples suivants illustrent deux façons d'utiliser la commande **asnqanalyze generate html report**.

- Générer un rapport HTML brut pour la base de données V95DB et les tables de contrôle sous le schéma ASN. Obtenez les informations des tables IBMQREP_SRC_COLS et IBMQREP_TRG_COLS, et également des tables de contrôle :

```
ASNQANALYZE GENERATE HTML REPORT DATABASE=V95DB
GETCOLS=ON GETMONITOR=ON -o qanalyzer_report.html
```

- Générer un rapport HTML brut pour les tables de contrôle de la base de données V95DB, sous le schéma ASNTEMP.

```
ASNQANALYZE GENERATE HTML REPORT DATABASE=v95db SCHEMA=ASNTEMP
```

(La sortie est envoyée vers le fichier V95DB.ASNTEMP.html)

asnqmfmt : mise en forme et visualisation des messages de réplication Q et de publication

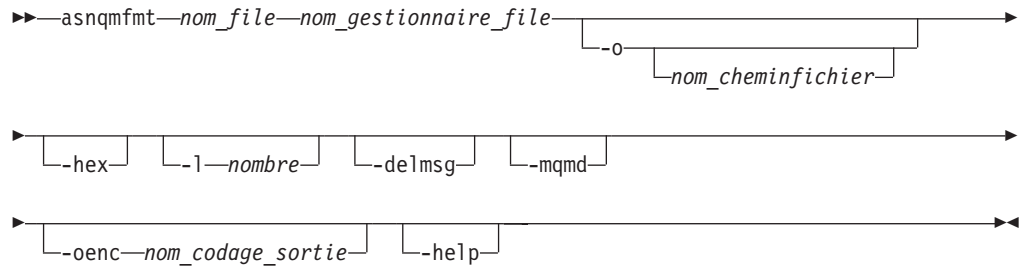
Utilisez la commande **asnqmfmt** pour mettre en forme et visualiser les messages utilisés dans la réplication Q et la publication d'événements . Exécutez cette commande sous Linux, UNIX, Windows ou UNIX System Services (USS) sous z/OS à l'invite du système d'exploitation ou dans un script d'interpréteur de commandes.

Important : Le format des messages de redémarrage de Q Capture a changé pour z/OS avec l'APAR PK78112 ou Linux, UNIX et Windows version 9.5 avec le groupe de correctifs 3. Vous ne pouvez pas utiliser un programme **asnqmfmt** antérieur à ce niveau pour formater des messages de ce nouveau format. De plus, un nouveau programme **asnqmfmt** ne peut pas formater les anciens messages de redémarrage.

z/OS Vous pouvez faire fonctionner le programme de formatage de messages avec le langage JCL. Vous pouvez trouver un échantillon de langage JCL dans qrhqual.SASNSAMP(ASNQMFMT).

Pour la publication d'événements, vous devez exécuter la commande **asnqmfmt** depuis le répertoire qui contient le fichier de définition de schéma mqcap.xsd. L'emplacement par défaut du fichier est SQLLIB/samples/rep1/q.

Syntaxe



Paramètres

Le tableau 56 décrit les paramètres d'appel pour **asnqmfmt** :

Tableau 56. définitions des paramètres d'appel *asnqmfmt*

Paramètre	Définition
<i>nom_file</i>	Définit le nom d'une file d'attente WebSphere MQ dont vous voulez formater, afficher et éventuellement supprimer les messages.
<i>nom_gestionnaire_file</i>	Définit le nom d'un gestionnaire de files d'attente WebSphere MQ où la file d'attente est définie.
-o <i>nom_cheminfichier</i>	Définit le nom du fichier contenant la sortie formatée. Si le paramètre -o n'est pas spécifié, les messages formatés seront écrits sur la sortie standard (stdout). Le fichier de sortie sera écrit sur un fichier z/OS si son nom commence par //. Par défaut, le fichier est créé dans le répertoire à partir duquel la commande asnqmfmt a été appelée. Vous pouvez modifier le répertoire en définissant un chemin avec le nom de fichier.
-hex	Spécifie que les messages sont formatés en hexadécimal. Si vous ne spécifiez pas ce paramètre, les messages seront affichés selon leur type de format de message, soit compact, délimité ou XML.
-l <i>nombre</i>	Spécifie le nombre de messages que vous voulez formater.
-delmsg	Spécifie que les messages seront supprimés de la file d'attente après leur formatage.
-mqmd	Spécifie que vous voulez afficher le descripteur de message WebSphere MQ pour chaque message formaté.
-oenc <i>nom_codage_sortie</i>	Spécifie une page de codes à utiliser pour le formatage des messages. Si vous ne spécifiez pas ce paramètre, les messages seront formatés dans la page de codes par défaut du système d'exploitation où la commande est appelée.
-help	Affiche les paramètres de la commande valide avec des descriptions.

Exemples pour asnqmfmt

Les exemples suivants illustrent comment utiliser la commande **asnqmfmt**.

Exemple 1

Pour afficher sur la sortie standard tous les messages qui sont dans la file d'attente d'envoi Q1 définie dans le gestionnaire de files d'attente QMGR1 :

```
asnqmfmt Q1 QMGR1
```

Exemple 2

Pour afficher tous les messages qui sont dans la file d'attente d'envoi Q1 dans un fichier appelé Q1_messages stocké dans le répertoire C:\qrepl(Windows) :

```
asnqmfmt Q1 QMGR1 -o C:\qrepl\Q1_messages
```

Exemple 3

Pour afficher sur la sortie standard une version hexadécimale de tous les messages qui sont dans la file d'attente d'administration ADMNQ1 définie dans le gestionnaire de files d'attente QMGR1 :

```
asnqmfmt ADMNQ1 QMGR1 -hex
```

Exemple 4

Pour afficher sur la sortie standard le corps du message et le descripteur de message de tous les messages qui sont dans la file d'attente d'administration ADMNQ1 définie dans le gestionnaire de files d'attente QMGR1, puis supprimer les messages de cette file d'attente :

```
asnqmfmt ADMNQ1 QMGR1 -delmsg -mqmd
```

Exemple 5

Pour afficher les 100 premiers messages qui sont dans la file d'attente de réception Q2 dans un fichier appelé Q2_messages stocké dans le répertoire C:\qrepl(Windows) :

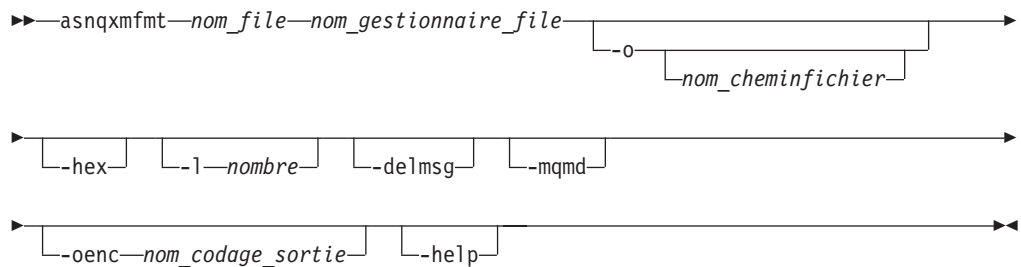
```
asnqmfmt Q2 QMGR2 -l 100 -o C:\qrepl\Q2_messages
```

asnqxfmt : formatage et affichage des messages de publication d'événements (z/OS)

Utilisez la commande **asnqxfmt** pour formater et afficher les messages délimités ou XML utilisés dans la publication d'événements. Cette commande s'exécute uniquement sur z/OS et UNIX System Services (USS) pour z/OS. La STEPLIB doit inclure les bibliothèques WebSphere MQ et XML Toolkit si elles ne sont pas installées dans le LNKLIST. Actuellement, **asnqxfmt** nécessite XML4C 1.4 version (HXML14A.SIXMMOD1).

z/OS Vous pouvez faire fonctionner le programme de formatage de messages avec le langage JCL. Vous pouvez trouver un échantillon de langage JCL dans qrhqlqual.SASNSAMP(ASNQXFMT).

Syntaxe



Paramètres

Le tableau 57 décrit les paramètres d'appel pour **asnqxfmt** :

Tableau 57. Définitions des paramètres d'appel *asnqxfmt*

Paramètre	Définition
<i>nom_file</i>	Définit le nom d'une file d'attente WebSphere MQ dont vous voulez formater, afficher et éventuellement supprimer les messages.
<i>nom_gestionnaire_file</i>	Définit le nom d'un gestionnaire de files d'attente WebSphere MQ où la file d'attente est définie.
-o <i>nom_cheminfichier</i>	Définit le nom du fichier contenant la sortie formatée. Si le paramètre -o n'est pas spécifié, les messages formatés seront écrits sur la sortie standard (stdout). Le fichier de sortie sera écrit sur un fichier z/OS si son nom commence par //. Par défaut, le fichier est créé dans le répertoire à partir duquel la commande asnqxfmt a été appelée. Vous pouvez modifier le répertoire en définissant un chemin avec le nom de fichier.
-hex	Spécifie que les messages sont formatés en hexadécimal. Si vous ne spécifiez pas ce paramètre, les messages seront affichés selon leur type de format de message, soit délimité soit XML.
-l <i>nombre</i>	Spécifie le nombre de messages que vous voulez formater.
-delmsg	Spécifie que les messages seront supprimés de la file d'attente après leur formatage.
-mqmd	Spécifie que vous voulez afficher le descripteur de message WebSphere MQ pour chaque message formaté.
-oenc <i>nom_codage_sortie</i>	Spécifie une page de codes à utiliser pour le formatage des messages. Si vous ne spécifiez pas ce paramètre, les messages seront formatés dans la page de codes par défaut du système d'exploitation où la commande est appelée.
-help	Affiche les paramètres de la commande valide avec des descriptions.

Exemples pour asnqxfmt

Les exemples suivants illustrent comment utiliser la commande **asnqxfmt**.

Exemple 1

Pour afficher sur la sortie standard tous les messages qui sont dans la file d'attente d'envoi Q1 définie dans le gestionnaire de files d'attente QMGR1 :

```
asnqxfmt Q1 QMGR1
```

Exemple 2

Pour afficher sur la sortie standard une version hexadécimale de tous les messages qui sont dans la file d'attente d'administration ADMNQ1 définie dans le gestionnaire de files d'attente QMGR1 :

```
asnqxfmt ADMNQ1 QMGR1 -hex
```

Exemple 3

Pour afficher sur la sortie standard le corps du message et le descripteur de message de tous les messages qui sont dans la file d'attente d'administration ADMNQ1 définie dans le gestionnaire de files d'attente QMGR1, puis supprimer les messages de cette file d'attente :

```
asnqxfmt ADMNQ1 QMGR1 -delmsg -mqmd
```

Chapitre 25. Tables de contrôle pour la réplication Q et la publication d'événements

Les tables de contrôle sont des tables de bases de données relationnelles utilisées pour stocker des informations pour les programmes de réplication Q et de publication d'événements. Ces tables de contrôle sont stockées sur le serveur Q Capture, sur le serveur Q Apply et sur le serveur de contrôle Monitor.

Tables de contrôle sur le serveur Q Capture

Les tables de contrôle sur le serveur Q Capture contiennent des informations sur les sources de données, les options des abonnements Q et des publications, les paramètres d'exploitation du programme Q Capture, les statistiques de performances Q Capture et d'autres métadonnées. Ces tables sont générées selon les options que vous avez définies dans les outils d'administration de la réplication.

Le tableau 58 décrit les tables de contrôle sur le serveur Q Capture.

Tableau 58. Tables de contrôle sur le serveur Q Capture

Nom de la table	Description
«Table IBMQREP_ADMINMSG», à la page 490	Une table interne contenant des messages d'administration reçus par un programme Q Capture.
«Table IBMQREP_CAPENQ», à la page 491	Permet de s'assurer qu'un seul programme Q Capture avec un schéma donné est en cours d'exécution par serveur Q Capture.
«Table IBMQREP_CAPENVINFO», à la page 492	Stocke la valeur des variables de l'environnement d'exécution et autres informations que les outils d'administration de la réplication utilisent pour accéder aux programmes distants.
«Table IBMQREP_CAPMON», à la page 492	Contient des statistiques sur les performances d'un programme Q Capture.
«Table IBMQREP_CAPPARMS», à la page 494	Contient les paramètres que vous avez définis pour contrôler les opérations d'un programme Q Capture.
«Table IBMQREP_CAPQMON», à la page 500	Contient des statistiques sur les performances d'un programme Q Capture pour chaque file d'attente d'envoi.
«Table IBMQREP_CAPTRACE», à la page 502	Contient des messages d'information, d'avertissement et d'erreur provenant d'un programme Q Capture.
«Table IBMQREP_COLVERSION», à la page 503	Permet aux programmes Q Capture et Capture de maintenir un suivi des différentes versions d'une table source.
«Table IBMQREP_EOLFLUSH», à la page 504	Une table interne qui force le vidage de la mémoire tampon de lecture du journal pour les sources Oracle.
«Table IBMQREP_EXCLSCHEMA», à la page 505	Contient des informations sur des tables qui sont exclues des abonnements Q de niveau schéma.
Table IBMQREP_IGNTRAN	Peut être utilisé pour informer le programme Q Capture sur les transactions que vous ne souhaitez pas capturer depuis le journal de récupération DB2 ou le journal redo Oracle.

Tableau 58. Tables de contrôle sur le serveur Q Capture (suite)

Nom de la table	Description
Table IBMQREP_IGNTRANTRC	Enregistre des informations sur les transactions qui ont été spécifiées à ignorer.
«Table IBMQREP_PART_HIST (Linux, UNIX, Windows)», à la page 507	Conserve un historique des changements sur les tables source partitionnées sur les systèmes Linux, UNIX, et Windows. Cette table est utilisée par la réplication Q et la réplication SQL.
«Table IBMQREP_SCHEMASUBS», à la page 509	Contient des informations sur des abonnements Q de niveau schéma, y compris la mappe de files de réplication qu'ils utilisent, le profil enregistré si utilisé et l'état.
«Table IBMQREP_SENDQUEUES», à la page 511	Contient des informations sur les files d'attente WebSphere MQ qu'un programme Q Capture utilise pour envoyer une transaction, un action sur une ligne, un objet de taille importante ou des messages d'information.
«Table IBMQREP_SIGNAL», à la page 515	Contient des signaux utilisés pour demander un programme Q Capture. Ces signaux sont insérés par un utilisateur, une application d'abonnement ou un programme Q Capture après qu'il a reçu un message de contrôle de la part du programme Q Apply ou d'une application d'abonnement.
«Table IBMQREP_SRC_COLS», à la page 520	Identifie des colonnes dans la table source qui sont répliquées ou publiées pour un abonnement Q ou une publication.
«Table IBMQREP_SRCH_COND», à la page 521	Une table interne qu'un programme Q Capture utilise pour évaluer la condition de recherche que vous avez définie pour un abonnement Q ou une publication.
«Table IBMQREP_SUBS», à la page 522	Contient des informations sur les abonnements Q et les publications, dont le type d'abonnement, les tables source, les conditions de recherche, les options d'envoi de données, les options de chargement de cible et les états.
«Table IBMQREP_TABVERSION», à la page 529	Permet aux programmes Q Capture et Capture de maintenir un suivi des différentes versions d'une table source.

Table IBMQREP_ADMINMSG

La table IBMQREP_ADMINMSG est une table interne qu'un programme Q Capture utilise pour enregistrer l'heure et l'identificateur des messages d'administration qu'il reçoit.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Clé primaire : MQMSGID

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 59 fournit une brève description des colonnes dans la table IBMQREP_ADMINMSG.

Tableau 59. Colonnes dans la table IBMQREP_ADMINMSG

Nom de colonne	Description
MQMSGID	Type de données : CHAR(24) FOR BIT DATA ; NULL admis : non Identificateur de message WebSphere MQ du message.
MSG_TIME	Type de données : TIMESTAMP ; NULL admis : non, avec valeur par défaut Horodatage sur le serveur Q Capture lorsque le message a été inséré dans cette table. Par défaut : horodatage actuel.

Table IBMQREP_CAPENQ

La table IBMQREP_CAPENQ permet de s'assurer que le schéma utilisé pour identifier le programme Q Capture et ses tables de contrôle est unique.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

La table IBMQREP_CAPENQ permet de s'assurer que :

- **Linux UNIX Windows** Pour DB2 pour Linux, UNIX et Windows, un seul programme Q Capture avec un schéma donné est en cours d'exécution par base de données.
- **z/OS** Pour le non partage de données de DB2 for z/OS, un seul programme Q Capture avec un schéma donné est en cours d'exécution par sous-système.
- **z/OS** Pour le partage de données de DB2 for z/OS, un seul programme Q Capture avec un schéma donné est en cours d'exécution par groupe de partage de données.

Lorsqu'un programme Q Capture est en cours d'exécution, il verrouille exclusivement cette table. Si le programme Q Capture démarre deux fois, la deuxième instance sera placée en attente sur verrouillage sur cette table. La table est créée vide.

Le tableau 60 fournit une brève description de la colonne dans la table IBMQREP_CAPENQ.

Tableau 60. Colonne dans la table IBMQREP_CAPENQ

Nom de colonne	Description
LOCKNAME	Type de données : INTEGER ; NULL admis : oui Cette colonne ne contient aucune donnée.

Table IBMQREP_CAPENVINFO

La table IBMQREP_CAPENVINFO contient huit lignes utilisées pour stocker la valeur des variables de l'environnement d'exécution et autres informations que les outils d'administration de la réplication utilisent pour accéder aux programmes distants.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 61 fournit une brève description des colonnes dans la table IBMQREP_CAPENVINFO.

Tableau 61. Colonnes dans la table IBMQREP_CAPENVINFO

Nom de colonne	Description
NAME	Type de données : VARCHAR(30) ; NULL admis : non STARTTIME Horodatage du démarrage du programme Q Capture. HOSTNAME Nom d'hôte TCP/IP du serveur lorsque le programme Q Capture est en cours d'exécution. LOGFILE Chemin d'accès et nom du fichier du fichier journal de diagnostic du programme Q Capture. TMPDIR Chemin d'accès au répertoire où se trouve la clé de communication interprocessus (IPC) du programme Q Capture. ASNUSEMQCLIENT Valeur de la variable d'environnement de réplication Q ASNUSEMQCLIENT . MQSERVER Valeur de la variable d'environnement WebSphere MQ MQSERVER . MQCHLLIB Valeur de la variable d'environnement WebSphere MQ MQCHLLIB . MQCHLTAB Valeur de la variable d'environnement WebSphere MQ MQCHLTAB .
VALUE	Type de données : VARCHAR(3800); NULL admis : Oui Pour chaque ligne de la table IBMQREP_CAPENVINFO, la colonne VALUE contient la valeur associée à la colonne NAME correspondante.

Table IBMQREP_CAPMON

Le programme Q Capture insère une ligne dans la table IBMQREP_CAPMON pour enregistrer des statistiques de performances pendant une période donnée. La valeur que vous définissez pour **MONITOR_INTERVAL** dans la table IBMQREP_CAPPARMS indique au programme Q Capture à quelle fréquence

insérer une ligne dans la table IBMQREP_CAPMON. La valeur MONITOR_LIMIT indique le nombre de minutes pendant lesquelles les lignes restent dans cette table avant d'être susceptibles d'être élaguées.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Index créé sans clause UNIQUE : MONITOR_TIME DESC

Important : N'utilisez pas SQL pour modifier cette table. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 62 fournit une brève description des colonnes dans la table IBMQREP_CAPMON.

Tableau 62. Colonnes dans la table IBMQREP_CAPMON

Nom de colonne	Description
MONITOR_TIME	Type de données : TIMESTAMP; Null admis : Non Horodatage sur le serveur Q Capture lorsque la ligne a été insérée dans cette table.
CURRENT_LOG_TIME	Type de données : TIMESTAMP; Null admis : Non Horodatage sur le serveur Q Capture de la dernière validation de la base de données vue par le programme de lecture de journal Q Capture.
CAPTURE_IDLE	Cette colonne est obsolète.
CURRENT_MEMORY	Type de données : INTEGER ; NULL admis : Non Quantité de mémoire (en octets) que le programme Q Capture a utilisée pour reconstruire des transactions à partir du journal.
ROWS_PROCESSED	Type de données : INTEGER ; NULL admis : Non Nombre de lignes (opérations d'insertion, de mise à jour ou de suppression individuelles) que le programme Q Capture a lues à partir du journal.
TRANS_SKIPPED	Type de données : INTEGER ; NULL admis : Non Nombre de transactions (contenant des lignes modifiées) qui n'ont pas été mises en file d'attente car les modifications ont été faites sur des colonnes qui ne font pas partie d'un abonnement Q ou d'une publication (le paramètre ALL_CHANGED_ROWS de la table IBMQREP_SUBS est défini sur la valeur Non par défaut).
TRANS_PROCESSED	Type de données : INTEGER ; NULL admis : Non Nombre de transactions que le programme Q Capture a traitées.
TRANS_SPILLED	Type de données : INTEGER ; NULL admis : Non Nombre de transactions que le programme Q Capture a déversées dans un fichier après dépassement du seuil MEMORY_LIMIT.
MAX_TRANS_SIZE	Type de données : INTEGER ; NULL admis : Non Transaction la plus grande, en octets, traitée par le programme Q.
QUEUES_IN_ERROR	Type de données : INTEGER ; NULL admis : Non Nombre de files d'attente qui n'ont pas accepté des messages.

Tableau 62. Colonnes dans la table IBMQREP_CAPMON (suite)

Nom de colonne	Description
RESTART_SEQ	<p>sources DB2 Type de données : VARCHAR(16) FOR BIT DATA ; Null admis : Non</p> <p>sources Oracle Type de données : RAW(10); Null admis : Non</p> <p>Le numéro d'ordre du journal à partir duquel le programme Q Capture démarre en mode de redémarrage à chaud. Cette valeur représente le premier LSN que le programme Q Capture a identifié pour lequel aucun enregistrement de validation ou d'abandon n'a encore été trouvé.</p>
CURRENT_SEQ	<p>sources DB2 Type de données : VARCHAR(16) FOR BIT DATA ; Null admis : Non</p> <p>sources Oracle Type de données : RAW(10); Null admis : Non</p> <p>Le numéro d'ordre du journal le plus récent dans le journal de reprise que le programme Q Capture lit.</p>
LAST_EOL_TIME	<p>Type de données : TIMESTAMP ; NULL admis : oui</p> <p>Heure sur le serveur de contrôle Q Capture à laquelle le programme Q Capture a atteint la fin du journal.</p>
LOGREAD_API_TIME	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Nombre de millisecondes que le programme Q Capture a passé à utiliser l'API de lecture du journal DB2 pour extraire les enregistrements de journaux.</p>
NUM_LOGREAD_CALLS	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Nombre d'appels d'API de lecture de journal effectués par Q Capture. Pour les sources Oracle, ce compteur affiche le nombre de fois où le programme Q Capture a extrait des enregistrements de journaux de l'utilitaire LogMiner.</p>
NUM_END_OF_LOGS	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Nombre de fois où Q Capture a atteint la fin du journal.</p>
LOGRDR_SLEEPTIME	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Nombre de millisecondes où l'unité d'exécution du lecteur de journal Q Capture était en veille car il n'y avait aucune modification à capturer ou car le programme Q Capture avait atteint sa limite de mémoire.</p>
MQCMIT_TIME	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Le nombre de millisecondes parmi l'intervalle du moniteur pendant lesquelles le programme Q Capture interagit avec l'interface de programme d'application de WebSphere MQ pour la validation des messages sur toutes les files d'attente d'envoi.</p>

Table IBMQREP_CAPPARMS

La table IBMQREP_CAPPARMS contient une seule ligne que vous pouvez modifier pour contrôler les opérations du programme Q Capture. Par exemple, vous pouvez paramétrer la méthode de traitement que le programme Q Capture utilise au démarrage. Vous pouvez également définir la durée d'attente du programme Q Capture avant qu'il valide les messages qui se trouvent dans des files d'attente d'envoi. Le programme Q Capture lit les modifications dans cette table uniquement au démarrage.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Index à entrées uniques : QMGR

Cette table contient des informations que vous pouvez mettre à jour à l'aide de SQL.

Important : si cette table n'a pas de ligne ou plusieurs lignes, le programme Q Capture ne s'exécutera pas.

Le tableau 63 fournit une brève description des colonnes dans la table IBMQREP_CAPPARMS.

Tableau 63. Colonnes dans la table IBMQREP_CAPPARMS

Nom de colonne	Description
QMGR	Type de données : VARCHAR(48) ; NULL admis : non Nom du gestionnaire de files d'attente WebSphere MQ que le programme Q Capture utilise.
REMOTE_SRC_SERVER	Type de données : VARCHAR(18) ; NULL admis : oui Réservé pour un usage ultérieur.
RESTARTQ	Type de données : VARCHAR(48) ; NULL admis : non Nom de la file d'attente qui stocke les messages de redémarrage pour le programme de lecture de journal Q Capture. Ce nom doit être unique pour chaque programme Q Capture.
ADMINQ	Type de données : VARCHAR(48) ; NULL admis : non Nom de la file d'attente qui reçoit des messages de contrôle provenant du programme Q Apply ou d'une application utilisateur. Ce nom doit être unique pour chaque programme Q Capture.

Tableau 63. Colonnes dans la table IBMQREP_CAPPARMS (suite)

Nom de colonne	Description
STARTMODE	<p>Type de données : VARCHAR(6) ; NULL admis : non, avec valeur par défaut</p> <p>Etapas effectuées par le programme Q Capture au démarrage.</p> <p>cold Le programme Q Capture vide les files d'attente de redémarrage et d'administration, et commence le traitement de tous les abonnements Q ou publications dont l'état est N (nouveau) ou A (actif). Avec un démarrage à froid, le programme Q Capture commence la lecture du journal de récupération DB2 ou du journal redo Oracle à la fin.</p> <p>warmsi (par défaut) Le programme Q Capture commence la lecture du journal à l'emplacement où il s'était arrêté, sauf s'il est démarré pour la première fois. Dans ce cas, le programme Q Capture effectue un démarrage à froid. Le mode de démarrage warsi garantit que le programme Q Capture démarre à froid uniquement lors de son démarrage initial.</p> <p>warmns Le programme Q Capture commence la lecture du journal à l'emplacement où il s'était arrêté. S'il ne peut pas être démarré à chaud, il ne passe pas à un démarrage à froid. Le mode de démarrage warmns empêche le programme Q Capture de démarrer à froid de façon intempestive. Dans le cas d'un démarrage à chaud, le programme Q Capture reprend le traitement où il s'était arrêté. Si des erreurs se produisent après le démarrage du programme Q Capture, le programme s'arrête sans modifier aucune table.</p> <p>Lors d'un démarrage à chaud, le programme Q Capture ne traitera que les abonnements Q ou les publications dont l'état n'est pas I (inactif).</p>
MEMORY_LIMIT	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Quantité de mémoire, en mégaoctets, que le programme Q Capture peut utiliser pour générer des transactions. Après l'utilisation de cette attribution, les transactions en mémoire se déversent dans un fichier. Valeur par défaut : 0 pour z/OS, 500 Mo pour Linux, UNIX, et Windows</p> <p>z/OS Une valeur de 0 indique au programme Q Capture de calculer une allocation de mémoire dérivée de la taille de la région de Q Capture dans JCL ou la tâche démarrée.</p>
COMMIT_INTERVAL	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Fréquence, en millisecondes, à laquelle le programme Q Capture émet un appel MQCMIT. Cet appel signale au gestionnaire de file d'attente MQ WebSphere qu'il doit rendre les messages de données et d'information qui ont été placés en attente accessibles au programme Q Apply ou aux applications utilisateur. Par défaut : 500</p>
AUTOSTOP	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant que le programme Q Capture s'arrête quand il atteint la fin de journal DB2 actif.</p> <p>Y Le programme Q Capture s'arrête quand il atteint la fin de journal DB2 actif.</p> <p>N (par défaut) Le programme Q Capture poursuit son exécution quand il atteint la fin de journal DB2 actif.</p>

Tableau 63. Colonnes dans la table IBMQREP_CAPPARMS (suite)

Nom de colonne	Description
MONITOR_INTERVAL	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Fréquence, en millisecondes, à laquelle le programme Q Capture ajoute des lignes à la table IBMQREP_CAPMON. Valeurs par défaut : 60000 millisecondes (1 minute) sous z/OS ; 30000 millisecondes (30 secondes) sous Linux, UNIX, et Windows</p>
MONITOR_LIMIT	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Nombre de minutes pendant lesquelles les lignes restent dans les tables IBMQREP_CAPMON et IBMQREP_CAPQMON avant d'être susceptibles d'être élaguées. A chaque intervalle d'élagage, des lignes de ces tables sont élaguées si elles sont plus anciennes que cette limite en fonction de l'horodatage actuel. Par défaut : 10080</p>
TRACE_LIMIT	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Nombre de minutes pendant lesquelles les lignes restent dans la table IBMQREP_CAPTRACE avant de pouvoir être élaguées. A chaque intervalle d'élagage, des lignes de la table IBMQREP_CAPTRACE sont élaguées si elles sont plus anciennes que cette limite en fonction de l'horodatage actuel. Par défaut : 10080</p>
SIGNAL_LIMIT	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Nombre de minutes pendant lesquelles les lignes restent dans la table IBMQREP_SIGNAL avant de pouvoir être élaguées. A chaque intervalle d'élagage, des lignes de la table IBMQREP_SIGNAL sont élaguées si elles sont plus anciennes que cette limite en fonction de l'horodatage actuel. Par défaut : 10080</p>
PRUNE_INTERVAL	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Fréquence, en secondes, à laquelle le programme Q Capture élague automatiquement des lignes dans les tables IBMQREP_CAPMON, IBMQREP_CAPQMON, IBMQREP_SIGNAL et IBMQREP_CAPTRACE. Par défaut : 300</p>
SLEEP_INTERVAL	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Nombre de millisecondes durant lesquelles le programme Q Capture reste en veille après le traitement du journal actif et des transactions restant éventuellement en mémoire. Valeurs par défaut : 500 millisecondes (0,5 secondes) pour les sources DB2 ; 2000 millisecondes (2 secondes) pour les sources Oracle</p>
LOGREUSE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si le programme Q Capture réutilise le fichier journal Q Capture ou fait des ajouts à ce journal.</p> <p>Y Au redémarrage, le programme Q Capture réutilise le fichier journal en nettoyant le fichier puis en écrivant dans le fichier vierge.</p> <p>N (par défaut) Le programme Q Capture ajoute de nouvelles informations à un fichier journal Q Capture existant lorsqu'il redémarre.</p>

Tableau 63. Colonnes dans la table IBMQREP_CAPPARMS (suite)

Nom de colonne	Description
LOGSTDOUT	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si le programme Q Capture envoie des messages de journal aux sorties autres que son fichier journal.</p> <p>Y Le programme Q Capture envoie des messages de journal à son fichier journal et à la console (stdout).</p> <p>N (par défaut) Le programme Q Capture dirige la plupart des messages de fichier journal vers le fichier journal uniquement.</p> <p>Les messages d'activation et de désactivation de l'initialisation, de l'arrêt et de l'abonnement vont vers la console (stdout) et vers le fichier journal.</p>
TERM	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant que le programme Q Capture s'arrête si la source DB2 ou le gestionnaire de files d'attente n'est pas disponible.</p> <p>Y (par défaut) Le programme Q Capture s'arrête si DB2 ou le gestionnaire de file d'attente est indisponible.</p> <p>N Le programme Q Capture continue de s'exécuter si DB2 ou le gestionnaire de file d'attente est indisponible. Lorsque DB2 ou le gestionnaire de files d'attente est disponible, Q Capture commence à envoyer des transactions là où il s'est arrêté sans qu'il soit nécessaire de redémarrer le programme.</p> <p>sources Oracle : Le paramètre term n'a pas d'effet sur les sources Oracle. La valeur par défaut est N pour la capture Oracle, ce qui indique que le programme Q Capture continue de s'exécuter si Oracle est indisponible.</p>
CAPTURE_PATH	<p>Type de données : VARCHAR(1040) ; NULL admis : oui</p> <p>Chemin où les fichiers créés par le programme Q Capture sont stockés.</p>
ARCH_LEVEL	<p>Type de données : CHAR(4) ; NULL admis : non, avec valeur par défaut</p> <p>Version des tables de contrôle. Pour la version 9.7 avec groupe de correctifs 3, la valeur est 0973. Les autres valeurs de ARCH_LEVEL sont 0802, 0901, 0905 et 0907.</p> <p>Avertissement : Au moment de la mise à jour de la table IBMQREP_CAPPARMS, ne modifiez pas la valeur dans cette colonne.</p>
COMPATIBILITY	<p>Type de données : CHAR(4) ; NULL admis : non, avec valeur par défaut</p> <p>Version des messages que le programme Q Capture envoie à un programme Q Apply.</p> <p>0907 (par défaut) Les messages de la version 9.7 sont envoyés. Cette valeur de compatibilité est également utilisée lorsque les serveurs source et cible sont au niveau de version 10 du serveur de réplication sous z/OS.</p> <p>0905 Les messages de la version 9.5 sont envoyés.</p> <p>0901 Les messages de la version 9.1 sont envoyés.</p> <p>0802 Les messages de la version 8 sont envoyés.</p>

Tableau 63. Colonnes dans la table IBMQREP_CAPPARMS (suite)

Nom de colonne	Description
LOB_SEND_OPTION	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant comment le programme Q Capture envoie les données LOB.</p> <p>I (par défaut) Intégré. Les valeurs LOB sont envoyées au sein du message de transaction. Les valeurs LOB intégrées peuvent améliorer les performances.</p> <p>S Séparé. Les valeurs LOB sont envoyées via un ou plusieurs messages LOB distincts qui suivent le message de transaction.</p>
QFULL_NUM_RETRIES	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Nombre de fois où le programme Q Capture essaiera un MQPUT pour la file d'attente d'envoi.</p> <p>Par défaut : 30 ; Maximum : 1000</p>
QFULL_RETRY_DELAY	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Durée pendant laquelle le programme Q Capture sera en veille avant le prochain essai. La valeur est spécifiée en millisecondes.</p> <p>Par défaut : 250 (millisecondes) ; minimum : 20 ; maximum : 3600000</p>
MSG_PERSISTENCE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indique si un programme Q Capture enregistre des messages permanents (consignés) ou non permanents (non consignés) dans des files d'attente WebSphere MQ.</p> <p>Y (par défaut) Q Capture enregistre des messages permanents sur toutes les files d'attente. Les messages sont consignés par le gestionnaire de files d'attente et peuvent être récupérés.</p> <p>N Q Capture enregistre des messages non permanents sur toutes les files d'attente. Les messages ne sont pas consignés et ne peuvent pas être récupérés.</p>
LOGRDBUFSZ	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Taille de la mémoire tampon (en Ko) que le programme Q Capture transmet à DB2 lorsque Q Capture extrait les enregistrements de journaux. DB2 remplit la mémoire tampon avec les enregistrements de journaux disponibles que Q Capture n'a pas extrait. Valeur par défaut : DB2 pour z/OS 66 Ko ; DB2 pour Linux, UNIX et Windows 256 Ko.</p>
CAPTURE_ALIAS	<p>Type de données : VARCHAR(8); Null admis : Oui</p> <p>Le nom d'alias pour la base de données ou le sous-système qui est utilisés comme serveur Q Capture. Il s'agit de l'alias tel que catalogué sur le système sur lequel s'exécutent les outils d'administration de la réplication et sont utilisés pour se connecter à la base de données source ou au sous-système pour créer des tables de contrôle Q Capture. Cette colonne est renseignée par le Centre de réplication ou le programme de ligne de commande ASNCLP lors de la création des tables de contrôle.</p>

Tableau 63. Colonnes dans la table IBMQREP_CAPPARMS (suite)

Nom de colonne	Description
STARTALLQ	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur qui signale au programme Q Capture s'il doit activer toutes les files d'attente d'envoi qui ne sont pas déjà actives lorsque Q Capture démarre. Les files d'attente d'envoi qui sont déjà actives sont toujours traitées lorsque Q Capture démarre.</p> <p>Y (par défaut) Lorsque le programme Q Capture démarre, il active toutes les files d'attente d'envoi qui ne sont pas déjà actives (A).</p> <p>N Lorsque le programme Q Capture démarre, il n'active pas toutes les files d'attente d'envoi qui sont inactives (I).</p>
WARNTXSZ	<p>Type de données : INTEGER; NULL admis : Non</p> <p>Indique si le programme Q Capture émet un message d'avertissement s'il rencontre des transactions plus volumineuses que la taille indiquée. Vous fournissez une valeur seuil en mégaoctets et les transactions qui dépassent le seuil provoquent l'affichage d'un message d'avertissement. Q Capture émet plusieurs messages d'avertissement si la taille de transaction est un multiple de la valeur WARNTXSZ. Par exemple, si vous définissez WARNTXSZ sur 10 Mo et que Q Capture rencontre une transaction de 30 Mo, trois avertissements sont envoyés (un pour 10 Mo, un pour 20 Mo et un pour 30 Mo).</p> <p>La valeur par défaut de 0 Mo signifie qu'aucun avertissement n'est jamais envoyé.</p>
LOGRD_ERROR_ACTION	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur qui spécifie l'action que le programme Q Capture doit mener en cas d'erreur de dictionnaire de compression pendant la lecture du journal de reprise. Une erreur "permanente" signifie que l'espace table a été réorganisée deux fois avec KEEPDICTIONARY=NO. Une erreur "transitoire" signifie toutes les autres erreurs du dictionnaire de compression.</p> <p>D (par défaut) Pour une erreur de dictionnaire permanente, désactivez l'abonnement Q. Pour une erreur de dictionnaire transitoire, arrêtez le programme Q Capture.</p> <p>E Pour l'un ou l'autre des types d'erreur, désactivez l'abonnement Q.</p> <p>S Pour l'un ou l'autre des types d'erreur, arrêtez le programme Q Capture.</p>

Table IBMQREP_CAPQMON

Serveur : serveur Q Capture

Schéma par défaut : ASN

Index créé sans clause UNIQUE : MONITOR_TIME

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le programme Q Capture insère des lignes dans la table IBMQREP_CAPQMON pour enregistrer des statistiques relatives à ses performances pour chaque file

d'attente d'envoi. La valeur que vous définissez pour `MONITOR_INTERVAL` dans la table `IBMQREP_CAPPARMS` indique à quelle fréquence le programme Q Capture effectue ces insertions. La valeur `MONITOR_LIMIT` paramètre le nombre de minutes pendant lesquelles les lignes restent dans la table `IBMQREP_CAPQMON` avant d'être susceptible d'être élaguées.

Le tableau 64 fournit une brève description des colonnes dans la table `IBMQREP_CAPQMON`.

Tableau 64. Colonnes dans la table `IBMQREP_CAPQMON`

Nom de colonne	Description
<code>MONITOR_TIME</code>	Type de données : <code>TIMESTAMP</code> ; Null admis : Non Horodatage sur le serveur Q Capture lorsque la ligne a été insérée dans cette table.
<code>SENDQ</code>	Type de données : <code>VARCHAR(97)</code> ; NULL admis : non Nom de la file d'attente d'envoi à laquelle cette ligne des statistiques de surveillance se rapporte.
<code>ROWS_PUBLISHED</code>	Type de données : <code>INTEGER</code> ; NULL admis : Non Nombre de lignes (opérations d'insertion, de mise à jour ou de suppression individuelles) que le programme Q Capture a mis dans cette file d'attente d'envoi.
<code>TRANS_PUBLISHED</code>	Type de données : <code>INTEGER</code> ; NULL admis : Non Nombre de transactions que le programme Q Capture a mis sur cette file d'attente d'envoi.
<code>CHG_ROWS_SKIPPED</code>	Type de données : <code>INTEGER</code> ; NULL admis : Non Nombre de lignes modifiées qui n'ont pas été mises en file d'attente d'envoi car les modifications ont été faites sur des colonnes qui ne font pas partie d'un abonnement Q ou d'une publication (le paramètre <code>ALL_CHANGED_ROWS</code> de la table <code>IBMQREP_SUBS</code> est défini sur la valeur Non par défaut).
<code>DELROWS_SUPPRESSED</code>	Type de données : <code>INTEGER</code> ; NULL admis : Non Nombre d'opérations de suppression de ligne qui n'ont pas été mises dans cette file d'attente d'envoi car l'abonnement Q ou la publication a été créé avec l'option d'élimination des lignes supprimées.
<code>ROWS_SKIPPED</code>	Type de données : <code>INTEGER</code> ; NULL admis : Non Nombre de lignes que le programme Q Capture n'a pas transmises à cette file d'attente d'envoi car elles ne satisfaisaient pas la condition de recherche définie dans l'abonnement Q ou la publication.
<code>LOBS_TOO_BIG</code>	Type de données : <code>INTEGER</code> ; NULL admis : Non Nombre de valeurs LOB qui ne passaient pas dans un message de transaction pour un intervalle de contrôle. Si l'action d'erreur de la file d'attente est définie sur E une valeur LOB vide est envoyée. Si l'action d'erreur de la file d'attente est définie sur S, le programme Q Capture s'arrête.
<code>XMLDOCS_TOO_BIG</code>	Type de données : <code>INTEGER</code> ; NULL admis : Non Nombre de documents XML qui ne passaient pas dans un message de transaction pour un intervalle de contrôle. Si l'action d'erreur de la file d'attente est définie sur E, un document XML fictif est inséré à la place. Si l'action d'erreur de la file d'attente est définie sur S, le programme Q Capture s'arrête.

Tableau 64. Colonnes dans la table IBMQREP_CAPQMON (suite)

Nom de colonne	Description
QFULL_ERROR_COUNT	Type de données : INTEGER; NULL admis : Non Nombre de fois où le programme Q Capture a essayé de mettre des messages (MQPUT) en file d'attente d'envoi spécifiée dans la colonne SENDQ.
MQ_BYTES	Type de données : INTEGER ; NULL admis : oui Nombre d'octets mis sur la file d'attente d'envoi lors de l'intervalle de surveillance, y compris les données provenant de la table source et de l'en-tête de message.
MQ_MESSAGES	Type de données : INTEGER ; NULL admis : oui Nombre de messages mis sur la file d'attente d'envoi lors de l'intervalle de surveillance.
CURRENT_SEQ	Type de données : VARCHAR(16) FOR BIT DATA ; Null admis : Oui Le numéro d'ordre du journal le plus récent dans le journal de reprise que le programme Q Capture lit pour cette file d'attente d'envoi. Q Capture met à jour cette colonne si la file d'attente d'envoi est active.
RESTART_SEQ	Type de données : VARCHAR(16) FOR BIT DATA ; Null admis : Oui Le numéro d'ordre du journal à partir duquel le programme Q Capture commence à placer des messages dans cette file d'attente d'envoi pendant un redémarrage à chaud. Cette valeur représente le premier numéro d'ordre du journal que le programme Q Capture a trouvé et qui n'a pas d'enregistrement de validation ou d'abandon. Q Capture met à jour cette colonne si la file d'attente d'envoi est active.
MQPUT_TIME	Type de données : INTEGER ; NULL admis : non, avec valeur par défaut Le nombre de millisecondes parmi l'intervalle du moniteur pendant lesquelles le programme Q Capture interagit avec l'interface de programme d'application de WebSphere MQ pour la mise en file d'attente des messages sur cette file d'attente d'envoi.

Table IBMQREP_CAPTRACE

La table IBMQREP_CAPTRACE contient des messages d'information, d'avertissement et d'erreur provenant du programme Q Capture.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Index créé sans clause UNIQUE : TRACE_TIME

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 65, à la page 503 fournit une brève description des colonnes dans la table IBMQREP_CAPTRACE.

Tableau 65. Colonnes dans la table IBMQREP_CAPTRACE

Nom de colonne	Description
OPERATION	<p>Type de données : CHAR(8) ; NULL admis : non</p> <p>Type de message provenant du programme Q Capture :</p> <p>INFOS Décrit les actions prises par le programme Q Capture.</p> <p>AVERTISSEMENT Décrit les conditions qui peuvent entraîner des erreurs pour le programme Q Capture.</p> <p>ERREUR Décrit les erreurs rencontrées par le programme Q Capture.</p>
TRACE_TIME	<p>Type de données : TIMESTAMP; Null admis : Non</p> <p>Heure sur le serveur Q Capture à laquelle le message a été mis dans la file d'attente d'envoi.</p>
DESCRIPTION	<p>Type de données : VARCHAR(1024); NULL admis : Non Type de données : INTEGER; Null admis : Oui</p> <p>Code anomalie pour le message d'erreur de réplication ou de publication d'événements.</p> <p>Identificateur de message ASN suivi par le texte de message. Cette colonne contient un texte en anglais uniquement.</p>
REASON_CODE	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Code anomalie pour le message d'erreur de réplication ou de publication.</p>
MQ_CODE	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Code anomalie pour le message d'erreur WebSphere MQ.</p>

Table IBMQREP_COLVERSION

La table IBMQREP_COLVERSION est utilisée par les programmes Q Capture et Capture pour conserver une trace des différentes versions d'une table source.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Index : LSN, TABLEID1, TABLEID2, POSITION

Index : TABLEID1 ASC, TABLEID2 ASC

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le programme Q Capture ou Capture insère des lignes dans cette table lorsque l'enregistrement ou l'abonnement Q pour une table source est activé pour la première fois, puis chaque fois que la table source est modifiée.

Le tableau 66, à la page 504 propose une brève description des colonnes de la table IBMQREP_COLVERSION.

Tableau 66. Colonnes de la table IBMQREP_COLVERSION

Nom de colonne	Description
LSN	Type de données : VARCHAR(16) FOR BIT DATA ; Null admis : Non Point du journal de récupération DB2 où le programme Q Capture ou Capture a détecté une nouvelle version de la table source.
TABLEID1	Type de données : SMALLINT ; NULL admis : non Identificateur d'objet (OBID) dans SYSIBM.SYSTABLES.
TABLEID2	Type de données : SMALLINT ; NULL admis : non Identificateur de base de données (DBID) dans SYSIBM.SYSTABLES.
POSITION	Type de données : SMALLINT ; NULL admis : non Position ordinale de la colonne dans la table, 0 désignant la première colonne dans la table.
NAME	Type de données : VARCHAR(128) ; NULL admis : non Nom de la colonne.
TYPE	Type de données : SMALLINT ; NULL admis : non Identificateur de type de donnée interne pour la colonne (SQLTYPE dans SYSIBM.SYSCOLUMNS).
LENGTH	Type de données : INTEGER ; NULL admis : Non Longueur maximale de donnée pour cette colonne.
NULLS	Type de données : CHAR(1); Null admis : Non Indicateur signalant si la colonne admet des valeurs nulles : Y La colonne admet des valeurs nulles. N La colonne n'admet pas de valeurs nulles.
DEFAULT	Type de données : VARCHAR(1536) ; NULL admis : oui Valeur par défaut de la colonne (DEFAULTVALUE) dans SYSIBM.SYSCOLUMNS. La valeur de cette colonne est NULL s'il n'y a pas de valeur par défaut.
CODEPAGE	Type de données : INTEGER ; NULL admis : oui Page de codes utilisée pour les données de cette colonne. La valeur est 0 si la colonne est définie en tant que FOR BIT DATA ou n'est pas de type chaîne. Valeur par défaut : NULL
SCALE	Type de données : INTEGER ; NULL admis : oui L'échelle des données décimales dans les colonnes décimales. La valeur est 0 pour les colonnes non décimales. Valeur par défaut : NULL

Table IBMQREP_EOLFLUSH

IBMQREP_EOLFLUSH est une table interne sur laquelle le programme Q Capture écrit lorsqu'Oracle LogMiner n'a pas répondu dans l'intervalle de temps spécifié par le paramètre **commit_interval**. Le fait d'écrire sur cette table permet de déterminer si des enregistrements de journaux mis en mémoire tampon ou non renvoyés sont disponibles pour être traités par Q Capture.

Serveur : serveur Q Capture pour les sources Oracle

Schéma par défaut : ASN

Important : N'utilisez pas SQL pour modifier cette table. La modification incorrecte de cette table peut avoir des résultats inattendus. Le format de cette table peut être modifié sans préavis.

La table suivante décrit la colonne de la table IBMQREP_EOLFLUSH.

Tableau 67. Colonne de la table IBMQREP_EOLFLUSH

Nom de colonne	Description
EOL_TIMEOUT	Type de données : TIMESTAMP; Null admis : Non Horodatage sur le serveur Q Capture lorsque la ligne a été insérée dans cette table.

Table IBMQREP_EXCLSCHEMA

La table IBMQREP_EXCLSCHEMA enregistre des informations à propos de tables qui sont exclues de l'abonnement Q au niveau schéma. Chaque ligne fournit les noms de table pour lesquelles des opérations SQL ne sont pas à répliquer même s'il existe un abonnement Q au niveau schéma dans la table IBMQREP_SCHEMASUBS. Par défaut, DB2 et les tables du catalogue de réplication sont exclues.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Important : ne modifiez pas cette table en utilisant SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

tableau 68 donne une brève description des colonnes de la table IBMQREP_EXCLSCHEMA.

Tableau 68. Colonnes de la table IBMQREP_EXCLSCHEMA

Nom de colonne	Description
SCHEMA_NAME	Type de données : VARCHAR(128) ; NULL admis : non Nom du schéma dans lequel certaines tables sont exclues d'un abonnement Q au niveau schéma. Les caractères génériques sont interdit dans cette colonne, à l'exception du caractère unique pourcentage (%) pour indiquer que les programmes de réplication doivent revoir tous les schémas pour des tables à exclure des abonnements Q au niveau schéma. z/OS : la valeur qui figure dans la colonne CREATOR de la table système SYSIBM.SYSTABLES pour les tables source qui sont exclues d'un abonnement Q de niveau schéma.
OBJECT_NAME	Type de données : VARCHAR(128) ; NULL admis : Non, avec valeur par défaut. Identifie les tables au sein d'un schéma qui sont exclues de la réplication, même s'il existe un abonnement Q de niveau schéma correspondant. Vous pouvez utiliser le signe pourcentage (%) comme caractère générique pour identifier plusieurs tables d'un ou de tous les schémas. Par défaut, DB2 et les tables du catalogue de réplication sont automatiquement exclues. Par défaut : %

Tableau 68. Colonnes de la table IBMQREP_EXCLSHEMA (suite)

Nom de colonne	Description
QMAPNAME	Type de données : VARCHAR(128) ; NULL admis : non La mappe de files de réplication utilisée pour un abonnement Q de niveau schéma.

Table IBMQREP_IGNTRAN

La table IBMQREP_IGNTRAN peut s'utiliser pour aviser le programme Capture des transactions que vous ne souhaitez pas capturer depuis le journal de récupération de DB2. Vous utilisez SQL pour insérer des lignes dans la table qui informe les programmes d'ignorer les transactions basées sur un ID d'autorisation, un jeton d'autorisation (z/OS uniquement), ou un nom de plan (z/OS uniquement).

Serveur : serveur Q Capture, serveur de contrôle Capture

Schéma par défaut : ASN

Index à entrées uniques : AUTHID ASC, AUTHTOKEN ASC, PLANNAME ASC

Le tableau 69 fournit une brève description des colonnes dans la table IBMQREP_IGNTRAN.

Tableau 69. Colonnes dans la table IBMQREP_IGNTRAN

Nom de colonne	Description
AUTHID	Type de données : CHAR(128) ; NULL admis : oui L'ID d'autorisation primaire pour la transaction que vous voulez ignorer.
AUTHTOKEN	Type de données : CHAR(30); Null admis : Oui z/OS Jeton d'autorisation (nom de travail) pour la transaction que vous voulez ignorer.
PLANNAME	Type de données : CHAR(8) ; NULL admis : oui z/OS Le nom de plan pour la transaction que vous voulez ignorer.
IGNTRANTRC	Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut Un indicateur informe le programme Q Capture ou Capture s'il faut tracer des transactions qui ont été ignorées en fonction de la valeur AUTHID, AUTHTOKEN, ou PLANNAME qui a été spécifiée dans la table IBMQREP_IGNTRAN : N (par défaut) La trace est désactivée. Y La trace est activée. Chaque fois qu'une transaction est ignorée, une ligne est insérée dans la table IBMQREP_IGNTRANTRC et un message est émis.

Table IBMQREP_IGNTRANTRC

La table EBMQREP_IGNTRANTRC enregistre des informations sur les transactions qui ont été spécifiées à ignorer.

Serveur : serveur Q Capture, serveur de contrôle Capture

Schéma par défaut : ASN

Index : IGNTRAN_TIME ASC

Important : Ne modifiez pas cette table en utilisant SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Une ligne est insérée dans la table IBMQREP_IGNTRANTRC lorsqu'une transaction est ignorée dans le journal de reprise DB2. Cette table est élaguée conformément au paramètre **trace_limit** pour le programme Q Capture ou Capture.

Le tableau 70 fournit une brève description des colonnes dans la table IBMQREP_IGNTRANTRC.

Tableau 70. Colonnes dans la table IBMQREP_IGNTRANTRC



Nom de colonne	Description
IGNTRAN_TIME	Type de données : TIMESTAMP ; NULL admis : non, avec valeur par défaut L'heure à laquelle la transaction a été ignorée. Valeur par défaut : horodatage actuel
AUTHID	Type de données : CHAR(128) ; NULL admis : oui L'ID d'autorisation primaire de la transaction qui a été ignoré.
AUTHTOKEN	Type de données : CHAR(30); Null admis : Oui  Jeton d'autorisation (nom de travail) pour la transaction qui a été ignorée.
PLANNAME	Type de données : CHAR(8) ; NULL admis : oui  Nom de plan pour la transaction qui a été ignorée.
TRANSID	Type de données : CHAR(10) POUR BIT DATA ; NULL admis : Non. Identificateur de transaction pour la transaction qui a été ignorée.
COMMITLSN	Type de données : CHAR(10) POUR BIT DATA ; NULL admis : Non. Numéro de séquence du journal de validation pour la transaction qui a été ignorée.

Table IBMQREP_PART_HIST (Linux, UNIX, Windows)

La table IBMQREP_PART_HIST conserve un historique des changements sur les tables source partitionnées sur les systèmes Linux, UNIX, et Windows. Cette table est utilisée par la réplication Q et la réplication SQL.

Les programmes Q Capture et Capture utilisent l'historique de partition pour les aider à traiter les changements de partition de données, tels que ajouter, joindre ou déconnecter. Une ligne, identifiée par un numéro d'ordre de journal (LSN), est insérée pour chaque partition dans la table source en deux occasions :

- Les premiers abonnements Q ou membre d'un ensemble d'abonnements de la table sont activés.
- La table est modifiée pour ajouter, joindre ou déconnecter une partition.

Les anciennes lignes de cette table sont élaguées en fonction de la valeur du paramètre **part_hist_limit**.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Clé principale : LSN, TABSCHEMA, TABNAME, DATAPARTITIONID, TBSPACEID, PARTITIONOBJECTID

Index : TABSCHEMA, TABNAME, LSN

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

tableau 71 donne une brève description des colonnes de la table IBMQREP_PART_HIST.

Tableau 71. Colonnes dans la table IBMQREP_PART_HIST

Nom de colonne	Description
LSN	<p>Type de données : VARCHAR(16) FOR BIT DATA; Null admis : Non</p> <p>Point du journal de récupération DB2 où le programme Q Capture ou Capture a détecté une nouvelle version de la table source. Ce numéro d'ordre de journal (LSN) exprime l'une des valeurs suivantes :</p> <ul style="list-style-type: none"> • Le LSN de l'insertion de signal CAPSTART pour le premier abonnement Q ou membre d'ensemble d'abonnements pour la table. • Pour les abonnements Q à l'état N (nouveau), la valeur restartLSN tirée du message de redémarrage de Q Capture, la valeur du paramètre lsn de Q Capture, ou le LSN obtenu de DB2 pour le paramètre Q Capture migrate=Y. Les abonnements Q à l'état N sont automatiquement démarrés lorsque Q Capture démarre. • Le LSN d'une opération ADD, ATTACH, ou DETACH sur une partition de table.
HISTORY_TIME	<p>Type de données : TIMESTAMP; Null admis : Non</p> <p>L'heure à laquelle Q Capture ou Capture a détecté de nouvelles informations de partition pour une table.</p>
TABSCHEMA	<p>Type de données : VARCHAR(128) ; NULL admis : non</p> <p>Le nom de schéma ou le qualificatif de haut niveau de la table source.</p>
TABNAME	<p>Type de données : VARCHAR(128) ; NULL admis : non</p> <p>Nom de la table source.</p>
DATAPARTITIONID	<p>Type de données : INTEGER; NULL admis : Non</p> <p>Identificateur de la partition de données. Cette valeur est lue dans la table système SYSCAT.DATAPARTITIONS.</p>
TBSPACEID	<p>Type de données : INTEGER; NULL admis : Non</p> <p>Identificateur de l'espace de table dans lequel sont stockées les données de table. Cette valeur est lue dans la table système SYSCAT.DATAPARTITIONS.</p>

Tableau 71. Colonnes dans la table IBMQREP_PART_HIST (suite)

Nom de colonne	Description
PARTITIONOBJECTID	Type de données : INTEGER; NULL admis : Non Identificateur de la partition de données dans l'espace de table. Cette valeur est lue dans la table système SYSCAT.DATAPARTITIONS.

Table IBMQREP_SCHEMASUBS

La table IBMQREP_SCHEMASUBS contient des informations sur des abonnements Q de niveau schéma, y compris la mappe de files de réplication qu'ils utilisent, le profil enregistré, si utilisé, et l'état.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Clé primaire : SCHEMA_SUBNAME

Important : ne modifiez pas cette table en utilisant SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 72 fournit une brève description des colonnes de la table IBMQREP_SCHEMASUBS.

Tableau 72. Colonnes de la table IBMQREP_SCHEMASUBS

Nom de colonne	Description
SCHEMA_SUBNAME	Type de données : VARCHAR(64) ; NULL admis : non Identificateur pour l'abonnement Q de niveau schéma.
QMAPNAME	Type de données : VARCHAR(128) ; NULL admis : non Nom de la mappe de files de réplication utilisée par l'abonnement Q de niveau schéma.
SCHEMA_NAME	Type de données : VARCHAR(128) ; NULL admis : non Identificateur relatif à un ensemble de schémas dont cet abonnement Q de niveau schéma fait partie. Cette colonne peut éventuellement contenir le signe de pourcentage (%) comme suffixe de caractère générique. (Le caractère unique "%" permet d'indiquer tous les schémas). z/OS : la valeur qui figure dans la colonne CREATOR de la table système SYSIBM.SYSTABLES pour les tables source à inclure dans l'abonnement Q de niveau schéma.
OBJECT_NAME	Type de données : VARCHAR(128) ; NULL admis : Non, avec valeur par défaut Identificateur relatif à un ensemble de tables qui appartiennent à cet abonnement Q de niveau schéma. Cette colonne peut éventuellement contenir le signe de pourcentage (%) comme suffixe de caractère générique. Par défaut : %

Tableau 72. Colonnes de la table IBMQREP_SCHEMASUBS (suite)

Nom de colonne	Description
SUBPROFNAME	<p>Type de données : VARCHAR(128) ; NULL admis : Non, avec valeur par défaut</p> <p>Identificateur relatif au profil enregistré qui permet d'indiquer des options pour l'abonnement Q de niveau table au sein de cet abonnement Q de niveau schéma. Valeur par défaut : ASNBIDI</p>
SUBGROUP	<p>Type de données : VARCHAR(30); Null admis : Oui</p> <p>Identificateur reliant tous les abonnements Q requis pour répliquer une table dans toutes les directions dans une configuration de réplication multidirectionnelle. Tous les abonnements Q faisant partie de la même configuration (par exemple, une configuration de réplication bidirectionnelle entre deux sites) doivent indiquer le même nom pour SUBGROUP pour tous les serveurs impliqués dans la configuration.</p>
SOURCE_NODE	<p>Type de données : SMALLINT ; NULL admis : non, avec valeur par défaut</p> <p>Un numéro d'identification pour le serveur source dans un abonnement Q bidirectionnel ou entre homologues. Par défaut : 0</p>
TARGET_NODE	<p>Type de données : SMALLINT ; NULL admis : non, avec valeur par défaut</p> <p>Un numéro d'identification pour le serveur cible dans un abonnement Q bidirectionnel ou entre homologues. Par défaut : 0</p>
STATE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Un indicateur est inséré par le programme Q Capture pour signaler l'état actuel de l'abonnement schema-level. L'état initial est N (nouveau). La zone STATE_INFO est initialement définie sur ASN7247I (nouvel abonnement schema-level).</p> <p>N (par défaut) L'abonnement schema-level est nouveau. Le programme Q Capture active automatiquement l'abonnement lorsque le programme est démarré ou réinitialisé. L'activation signifie que les opérations SQL telles que CREATE TABLE et DROP TABLE sont répliquées pour les tables qui correspondent au modèle de l'abonnement.</p> <p>I L'abonnement schema-level est inactif. Le programme Q Capture a vu un signal STOP_SCHEMASUB dans le journal ou une erreur est survenue alors que l'abonnement schema-level était désactivé. Le programme Q Capture a cessé d'envoyer des messages pour cet abonnement mais a continué avec les autres.</p> <p>A L'abonnement schema-level est actif. Le programme Q Capture envoie des messages en fonction des options définies pour l'abonnement.</p>
STATE_TIME	<p>Type de données : TIMESTAMP ; NULL admis : non, avec valeur par défaut</p> <p>L'horodatage du dernier changement de l'état de l'abonnement Q de niveau schéma. Valeur par défaut : horodatage actuel</p>
STATE_INFO	<p>Type de données : CHAR(8) ; NULL admis : oui</p> <p>L'identificateur associé au message qui a été émis à propos de l'état de l'abonnement Q de niveau schéma. Pour plus d'informations, voir la table IBMQREP_CAPTRACE ou le journal de diagnostic Q Capture.</p>

Table IBMQREP_SENDQUEUES

La table IBMQREP_SENDQUEUES contient des informations sur les files d'attente WebSphere MQ qu'utilise un programme Q Capture pour envoyer des données et des messages d'information. Chaque instance du programme Q Capture peut fonctionner avec plusieurs files d'attente d'envoi. Chaque file d'attente d'envoi est définie de manière unique dans la table IBMQREP_SENDQUEUES.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Clé primaire : SENDQ

Index à entrées uniques : PUBQMAPNAME

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 73 fournit une brève description des colonnes dans la table IBMQREP_SENDQUEUES.

Tableau 73. Colonnes dans la table IBMQREP_SENDQUEUES

Nom de colonne	Description
PUBQMAPNAME	Type de données : VARCHAR(128) ; NULL admis : Non Nom de la mappe de files pour la publication qui comprend cette file d'attente d'envoi. Pour les abonnements Q, ce nom doit correspondre à la valeur REPQMAPNAME dans la table IBMQREP_RECVQUEUES.
SENDQ	Type de données : VARCHAR(48) ; NULL admis : non Nom unique pour cette file d'attente d'envoi. Le nom peut représenter la définition locale d'une file d'attente distante ou une file d'attente locale. Les noms de files d'attente ne peuvent pas contenir de blancs.
RECVQ	Type de données : VARCHAR(48) ; NULL admis : oui Nom de la file d'attente de réception pour cet abonnement Q. Il s'agit d'une file d'attente locale sur le serveur Q Apply. Les noms de files d'attente ne peuvent pas contenir de blancs.
DESCRIPTION	Type de données : VARCHAR(254) ; NULL admis : oui Description fournie par l'utilisateur de la mappe de files de publication contenant cette file d'attente d'envoi.

Tableau 73. Colonnes dans la table IBMQREP_SENDQUEUES (suite)

Nom de colonne	Description
MESSAGE_FORMAT	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Format utilisé pour coder les messages qui sont mis en file d'attente d'envoi.</p> <p>C (par défaut) Le programme Q Capture code les transactions de la base de données source dans un format compact que lit le programme Q Apply.</p> <p>X Le programme Q Capture encode les transactions de la base de données source au format XML (eXtensible Markup Language).</p> <p>J Le programme Q Capture code les transactions de la base de données source dans un format XML (Extensible Markup Language) avec un en-tête MQRFH2.</p> <p>D Le programme Q Capture encode les transactions de la base de données source dans un format délimité.</p>
MSG_CONTENT_TYPE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si les messages mis en file d'attente contiendront une transaction de base de données complète ou une opération de ligne uniquement.</p> <p>T (par défaut)</p> <p>R Les messages contiennent une seule opération de mise à jour, d'insertion ou de suppression et des informations sur la transaction à laquelle elle appartient.</p>
STATE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur inséré par le programme Q Capture pour montrer l'état de la file d'attente d'envoi.</p> <p>A (par défaut) Active. Les transactions sont en cours d'écriture dans cette file d'attente.</p> <p>I Inactive. Une erreur grave a été rencontrée dans la file d'attente.</p>
STATE_TIME	<p>Type de données : TIMESTAMP ; NULL admis : non, avec valeur par défaut</p> <p>Horodatage sur le serveur Q Capture de la dernière modification de l'état de la file d'attente d'envoi. Valeur par défaut : horodatage Current</p>
STATE_INFO	<p>Type de données : CHAR(8) ; NULL admis : oui</p> <p>Numéro du message ASN relatif à l'état de la file d'attente. Pour plus de détails, voir la table IBMQREP_CAPTRACE ou le journal de diagnostic Q Capture.</p>

Tableau 73. Colonnes dans la table *IBMQREP_SENDQUEUES* (suite)

Nom de colonne	Description
ERROR_ACTION	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur qui signale au programme Q Capture ce qu'il doit faire si la file d'attente d'envoi n'accepte plus les messages. Par exemple, la file d'attente peut être pleine ou le gestionnaire de files d'attente a peut-être signalé une erreur grave pour cette file d'attente.</p> <p>S (par défaut) Le programme Q Capture s'arrête.</p> <p>I Cette valeur est obsolète et traitée comme s'il s'agissait de la valeur S.</p> <p>Q Le programme Q Capture poursuit l'insertion des messages dans d'autres files d'attente d'envoi, mais arrête l'insertion de messages dans cette file d'attente et définit son état sur Inactive (I). Q Capture laisse également l'état de tous les abonnements Q ou publications spécifiant cette file d'attente sur Active (A).</p> <p>Après avoir corrigé l'erreur de la file d'attente d'envoi, vous devez effectuer l'une des opérations suivantes :</p> <ul style="list-style-type: none"> • Emettre une commande startq pour indiquer à Q Capture de démarrer l'insertion des messages dans la file d'attente (ce qui définit son état sur Active). • Redémarrer le programme Q Capture. • Utiliser la commande Q Capture reinit pour recharger tous les abonnements Q à partir des tables de contrôle Q Capture.
LOB_TOO_BIG_ACTION	<p>Indicateur signalant au programme Q Capture ce qu'il doit faire si une seule valeur d'objet LOB dépasse la taille de message maximale autorisée pour cette file d'attente d'envoi.</p> <p>Q (par défaut) Le programme Q Capture suit l'action d'erreur définie pour la file d'attente d'envoi.</p> <p>E Le programme Q Capture envoie des valeurs LOB vides si les données ne tiennent pas dans un seul message de transaction. Si la valeur de remplacement est trop volumineuse pour un message, Q Capture suit l'action d'erreur pour la file d'attente.</p>
XML_TOO_BIG_ACTION	<p>Indicateur signalant au programme Q Capture ce qu'il doit faire si un seul document XML dépasse la taille de message maximale autorisée pour cette file d'attente d'envoi.</p> <p>Q (par défaut) Le programme Q Capture suit l'action d'erreur définie pour la file d'attente d'envoi.</p> <p>E Le programme Q Capture envoie une marque de réservation XML. Si la marque de réservation XML est trop volumineuse pour un message, Q Capture suit l'action d'erreur pour la file d'attente.</p>

Tableau 73. Colonnes dans la table IBMQREP_SENDQUEUES (suite)

Nom de colonne	Description
HEARTBEAT_INTERVAL	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Fréquence, en secondes, à laquelle le programme Q Capture envoie des messages sur cette file d'attente pour signaler au programme Q Apply ou à une application utilisateur que le programme Q Capture est toujours en cours d'exécution quand il n'y a pas de modification à publier. La valeur doit être un multiple de la valeur COMMIT_INTERVAL ou sera arrondie au multiple le plus proche (COMMIT_INTERVAL est paramétré dans la table IBMQREP_CAPPARMS). Une valeur de 0 (valeur par défaut) indique que le programme Q Capture n'envoie pas de messages heartbeat.</p> <p>Pour les messages délimités (MESSAGE_FORMAT=D), la valeur de HEARTBEAT_INTERVAL doit être 0.</p> <p>Remarque : La valeur HEARTBEAT_INTERVAL définie dans la table IBMQREP_SENDQUEUES est différente du paramètre HBINT (intervalle des pulsations) que vous pouvez définir pour un canal WebSphere MQ. Pour plus d'informations, voir <i>WebSphere MQ Script (MQSC) Command Reference</i>.</p>
MAX_MESSAGE_SIZE	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Taille maximale (en kilo-octets) de la mémoire tampon utilisée pour envoyer les messages vers cette file d'attente d'envoi. La taille de la mémoire tampon ne doit pas être supérieure à l'attribut de la longueur de message maximale (MAXMSGL) de WebSphere MQ défini pour les files d'attente qui contiendront le message, sinon tous les abonnements Q et publications utilisant cette file d'attente d'envoi seront invalidés. Par défaut : 64 Ko</p>
APPLY_SERVER	<p>Type de données : VARCHAR(18) ; NULL admis : oui</p> <p>Nom de la base de données où les exécutions et les cibles du programme Q Apply sont définies. Pour z/OS, il s'agit d'un nom d'emplacement.</p>
APPLY_ALIAS	<p>Type de données : VARCHAR(8); Null admis : Oui</p> <p>Alias de la base de données DB2 qui correspond au serveur Q Apply nommé dans la colonne APPLY_SERVER.</p>
APPLY_SCHEMA	<p>Type de données : VARCHAR(128) ; NULL admis : Oui</p> <p>Schéma du programme Q Apply appliquant les transactions de cette file d'attente d'envoi.</p>
MESSAGE_CODEPAGE	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Page de codes utilisée pour coder les messages pour la publication d'événements.</p>
COLUMN_DELIMITER	<p>Type de données : CHAR(1) ; NULL admis : oui</p> <p>Caractère unique utilisé pour séparer les entrées d'en-tête et les données de colonne au sein d'un message délimité. Par défaut : virgule (,)</p>
STRING_DELIMITER	<p>Type de données : CHAR(1) ; NULL admis : oui</p> <p>Caractère unique utilisé pour encadrer les données de chaîne dans un message délimité. Par défaut : guillemets (")</p>
RECORD_DELIMITER	<p>Type de données : CHAR(1) ; NULL admis : oui</p> <p>Caractère unique utilisé pour séparer les enregistrements de données modifiées au sein d'un message délimité. Par défaut : caractère de nouvelle ligne</p>

Tableau 73. Colonnes dans la table IBMQREP_SENDQUEUES (suite)

Nom de colonne	Description
DECIMAL_POINT	<p>Type de données : CHAR(1) ; NULL admis : oui</p> <p>Caractère unique utilisé pour séparer la partie fractionnelle de données numériques dans un message délimité. Par défaut : point (.)</p>
SENDRAW_IFERROR	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Pour la publication d'événements : indique si les données de type caractères sont publiées lorsque ces données causent une erreur de conversion de page de code. Un exemple de données de type caractères non valides est un caractère codé sur deux octets incomplet et un caractère codé manquant ('0F'x) dans une chaîne stockée dans une page de code en japonais ou coréen. La chaîne suivante doit avoir '830F' à la fin mais ce n'est pas le cas : x'0E4481448244'.</p> <p>N (par défaut) Aucune donnée n'est envoyée pour la zone de caractère qui n'a pas réussi la conversion de la page de code. Pour le format de message délimité, aucune des données des colonnes de caractères dans la ligne n'est envoyée lorsqu'une zone à caractère unique échoue dans la conversion de la page de code. A la place, toutes les colonnes de caractères sont envoyées comme valeurs null et un indicateur au début de la ligne est défini sur IBM-INVALID-COLUMN-####-@-NULL, où #### représente la première colonne dans laquelle le programme Q Capture a détecté des données non valides et @ est soit B (avant la valeur) soit A (après la valeur). Tous les objets LOB ou colonnes XML sont envoyés en tant que valeurs null.</p> <p>Y Les données de type caractère sont envoyées sous un format de chaîne hexadécimal. Pour les messages délimités, la zone identifiée est définie sur IBM-INVALID-COLUMN-####-@-HEX, et toutes les colonnes de caractères de la ligne sont envoyées sous un format de chaîne hexadécimal. La zone #### représente la première colonne dans laquelle le programme Q Capture a détecté des données non valides et @ est soit B (avant la valeur) soit A (après la valeur). Toutes les colonnes d'objet LOB ou XML sont envoyées en tant que valeurs null.</p>

Table IBMQREP_SIGNAL

La table IBMQREP_SIGNAL permet à un utilisateur, à une application utilisateur ou au programme Q Apply de communiquer avec un programme Q Capture.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Cette table contient des informations que vous pouvez mettre à jour à l'aide de SQL.

Un utilisateur ou une application utilisateur peut insérer des lignes dans la table IBMQREP_SIGNAL pour demander que le programme Q Capture commence la capture des modifications issues du journal pour une table source ou effectue d'autres actions telles que désactiver d'un abonnement Q ou ignorer une transaction. Le programme Q Apply ou une application utilisateur peut réaliser les mêmes requêtes en envoyant des messages de contrôle au programme Q Capture qui insère ensuite les signaux correspondants dans la table IBMQREP_SIGNAL. Le programme Q Capture reçoit les signaux quand il lit l'enregistrement de journal pour l'insertion dans la table IBMQREP_SIGNAL.

Les enregistrements dans cette table avec une valeur SIGNAL_STATE égale à C (complet) ou des enregistrements avec un horodatage admissible pour l'élagage sont supprimés quand le programme Q Capture réalise un élagage.

Le tableau 74 fournit une brève description des colonnes dans la table IBMQREP_SIGNAL.

Tableau 74. Colonnes dans la table IBMQREP_SIGNAL

Nom de colonne	Description
SIGNAL_TIME	<p>Type de données : TIMESTAMP ; NULL admis : non, avec valeur par défaut</p> <p>Un horodatage utilisé pour identifier de façon unique la ligne. Le programme Q Capture utilise cette valeur pour trouver la ligne correcte dans la table des signaux indiquant la fin du traitement du signal Q Capture. Valeur par défaut : horodatage actuel</p>
SIGNAL_TYPE	<p>Type de données : VARCHAR(30) ; NULL admis : non</p> <p>Indicateur signalant le type de signal qui a été posté :</p> <p>CMD Une ligne insérée par les commandes d'administration, asnjccmd, le Centre de réplication ou une autre application. Voir la colonne SIGNAL_SUBTYPE pour une liste des sous-types de signaux disponibles.</p> <p>USER Un signal posté par un utilisateur. Le programme Q Capture met à jour la colonne SIGNAL_LSN avec le numéro d'ordre du journal correspondant au moment où le signal a été inséré. Le programme Q Capture met aussi à jour en passant la valeur dans la colonne SIGNAL_STATE de attente (pending) (P) à reçu (received) (R).</p>
SIGNAL_SUBTYPE	<p>Type de données : VARCHAR(30); Null admis : Oui</p> <p>Le type d'action qu'un signal de type CMD demande au programme Q Capture d'effectuer.</p> <p>CAPSTART Démarrer la capture des modifications pour un abonnement Q ou une publication.</p> <p>CAPSTOP Arrêter la capture des modifications pour un abonnement Q ou une publication.</p> <p>QINERROR Exécuter l'action d'erreur définie pour la file d'attente d'envoi dans la table IBMQREP_SENDQUEUES.</p> <p>LOADDONE Confirmer la réception du signal provenant du programme Q Apply ou de l'application utilisateur. Le signal LOADDONE indique au programme Q Capture que la table cible est chargée.</p> <p>STOP Arrêter les modifications de capture et s'arrêter.</p> <p>IGNORETRANS Ignorer la transaction DB2 contenant ce signal.</p> <p>REINIT_SUB Désactiver puis activer un abonnement Q ou une publication à l'aide des dernières valeurs dans les tables IBMQREP_SUBS, IBMQREP_SRC_COLS et IBMQREP_SENDQUEUES. Ce signal ne demande pas un nouveau chargement des cibles.</p>

Tableau 74. Colonnes dans la table IBMQREP_SIGNAL (suite)

Nom de colonne	Description
SIGNAL_SUBTYPE (suite)	<p>ADDCOL Ajouter une colonne à un abonnement actif et unidirectionnel Q ou à une publication.</p> <p>STARTQ Commencer à placer des messages dans une file d'attente définie ou dans toutes les files d'attente inactives.</p> <p>STOPQ Arrête de placer des messages dans une file d'attente définie ou dans toutes les files d'attente inactives.</p> <p>P2PNEW2MEMB Un signal interne utilisé pour initialiser un abonnement Q entre homologues. Le signal est inséré sur un nouveau serveur et contient le nombre de serveurs actifs dans la configuration entre homologues.</p> <p>P2PMEMB2NEW Un signal interne utilisé pour initialiser un abonnement Q entre homologues. Le signal est inséré sur des serveurs actifs.</p> <p>P2PMEMB2INIT Un signal interne utilisé pour initialiser un abonnement Q entre homologues. Le signal est inséré sur des serveurs actifs.</p> <p>P2PSPOOLING Un signal interne utilisé pour initialiser un abonnement Q entre homologues. Le signal est inséré sur le serveur initiant un nouvel abonnement.</p> <p>P2PLOADDONE Un signal interne utilisé pour initialiser un abonnement Q entre homologues. Le signal est inséré sur un nouveau serveur.</p> <p>P2PSUBSTOP Un signal interne utilisé pour désactiver un abonnement Q entre homologues. Le signal est inséré sur le serveur qui est désactivé.</p> <p>P2PSUBSTOPPING Un signal interne utilisé pour désactiver un abonnement Q entre homologues. Le signal est inséré sur les serveurs actifs restants.</p> <p>P2PREADYTOSTOP Un signal interne utilisé pour désactiver un abonnement Q entre homologues. Le signal est inséré sur le serveur qui est désactivé.</p> <p>P2PNORECAPTURE Un signal inséré par le programme Q Apply pour empêcher le programme Q Capture de réenregistrer les modifications. Utilisé en réplication bidirectionnelle. Remarque : Les signaux P2PNORECAPTURE sont élagués conformément au paramètre prune_interval, contrairement aux autres sous-types de signaux, qui sont élagués conformément au paramètre signal_limit.</p> <p>REPLICATE_LOAD Signal interne utilisé par les programmes Q Capture et Q Apply lorsqu'ils répliquent des opérations de chargement de la table source. Ce signal peut également être utilisé lorsque le programme Q Apply modifie un état d'abonnement Q, lorsqu'il traite un signal ADDCOL ou une commande reinit par exemple.</p>

Tableau 74. Colonnes dans la table IBMQREP_SIGNAL (suite)

Nom de colonne	Description
SIGNAL_SUBTYPE	<p>P2PCREATESUB Signal interne qui est inséré par le programme Q Apply lors de la réplication d'une opération CREATE TABLE dans une configuration de réplication multidirectionnelle. Le signal informe le programme Q Capture apparié au niveau de son serveur qu'un abonnement Q est à créer dans l'autre direction.</p> <p>P2PDROPSUB Signal interne qui est inséré par le programme Q Apply lors de la réplication d'une opération DROP TABLE dans une configuration de réplication multidirectionnelle. Le signal informe le programme Q Capture apparié au niveau de son serveur qu'un abonnement Q est à supprimer dans l'autre direction.</p> <p>START_SCHEMASUB Signal qui donne l'instruction au programme Q Capture de démarrer la capture d'opérations SQL comme CREATE et DROP pour des tables dans un schéma spécifié.</p> <p>STOP_SCHEMASUB Signal qui donne l'instruction au programme Q Capture d'arrêter la capture des opérations SQL comme CREATE et DROP pour des tables dans un schéma spécifié.</p> <p>REINIT_SCHEMASUB Signal qui donne l'instruction au programme Q Capture de recharger tous les changements ou ajouts effectués dans la table IBMQREP_SUBS_PROF pour les profils relatifs à des abonnements Q de niveau schéma. Ces changements ne s'appliquent qu'aux nouvelles tables (et leurs abonnements Q) qui ont été créées dans le schéma spécifié.</p>

Tableau 74. Colonnes dans la table IBMQREP_SIGNAL (suite)

Nom de colonne	Description
SIGNAL_INPUT_IN	<p>Type de données : VARCHAR(500) ; NULL admis : oui</p> <p>Si SIGNAL_TYPE=USER, la colonne contient une entrée définie par l'utilisateur. Si SIGNAL_TYPE=CMD, cette valeur dépend de la valeur SIGNAL_SUBTYPE :</p> <p>CMD + CAPSTART Nom de l'abonnement Q ou de la publication.</p> <p>CMD + CAPSTOP Nom de l'abonnement Q ou de la publication.</p> <p>CMD + LOADDONE Nom de l'abonnement Q ou de la publication.</p> <p>CMD + STOP NULL (aucune valeur requise). Vous pouvez indiquer CAPTUREUPTO pour demander à Q Capture d'arrêter de capturer les modifications validées à ou avant un horodatage fourni, puis de s'arrêter. Vous pouvez également indiquer STOPAFTER pour demander à Q Capture d'arrêter de capturer les modifications immédiatement, puis de s'arrêter lorsque tous les messages sont supprimés de la file d'attente de transmission (ou d'une file d'attente d'envoi-réception locale partagée) ou après que toutes les modifications atteignant le point d'arrêt soient appliquées à la cible. Pour plus d'informations, voir «Arrêt d'un programme Q Capture à un point spécifié», à la page 264.</p> <p>CMD + IGNORETRANS NULL (aucune valeur requise).</p> <p>CMD + QINERROR Pour une application utilisateur, nom de la file d'attente d'envoi en erreur. Pour le programme Q Apply, nom de la file d'attente d'envoi en erreur et les jetons des numéros de message ASN séparés par un espace.</p> <p>CMD + REINIT_SUB Nom de l'abonnement Q ou de la publication.</p> <p>CMD + ADDCOL Nom de l'abonnement Q ou de la publication et nom de la colonne de table source ajoutés, séparés par un point-virgule. Utilisez le format 'subname;nom_colonne;nom_colonne_source;nom_colonne_cible'. Suivez ces exemples :</p> <p>Ajout de colonne dans la table source pour l'abonnement Q et la table cible QSUB1;COL10</p> <p>Ajout de colonne et d'image-avant de la colonne (pour des tables cibles CCD) QSUB1;COL10;XCOL10</p> <p>Ajout de colonne sans image-avant mais avec un nom de colonne cible différent QSUB1;COL10;;TRGCOL10 (placez deux points-virgules (;) pour indiquer que vous omettez la colonne d'image-avant.)</p> <p>CMD + STARTQ Nom de la file d'attente ou ALL.</p> <p>CMD + STOPQ Nom de la file d'attente ou ALL.</p>

Tableau 74. Colonnes dans la table *IBMQREP_SIGNAL* (suite)

Nom de colonne	Description
SIGNAL_INPUT_IN (suite)	<p>CMD + START_SCHEMASUB Nom de l'abonnement de niveau schéma.</p> <p>CMD + STOP_SCHEMASUB Nom de l'abonnement de niveau schéma.</p> <p>CMD + REINIT_SCHEMASUB Nom de l'abonnement de niveau schéma.</p>
SIGNAL_STATE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant l'état du signal.</p> <p>P (par défaut) Le signal est en attente ; le programme Q Capture ne l'a pas encore reçu.</p> <p>R Le programme Q Capture a reçu le signal.</p> <p>C Le programme Q Capture a terminé le traitement du signal.</p> <p>F Le signal a échoué. Par exemple, le programme Q Capture ne peut pas effectuer CAPSTART car un abonnement Q ou une publication est incorrect(e).</p>
SIGNAL_LSN	<p>Type de données : VARCHAR(16) FOR BIT DATA ; NULL admis : Yes</p> <p>Numéro d'ordre du journal logique de l'enregistrement de journal pour l'insertion dans la table SIGNAL.</p>

Table **IBMQREP_SRC_COLS**

La table *IBMQREP_SRC_COLS* répertorie les colonnes de la table source pour lesquelles les modifications doivent être capturées.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Clé primaire : SUBNAME, SRC_COLNAME

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 75 fournit une brève description des colonnes dans la table *IBMQREP_SRC_COLS*.

Tableau 75. Colonnes dans la table *IBMQREP_SRC_COLS*

Nom de colonne	Description
SUBNAME	<p>Type de données : VARCHAR(30) ; NULL admis : non</p> <p>Nom de l'abonnement Q ou de la publication pour cette table source.</p>
SRC_COLNAME	<p>Type de données : VARCHAR(128); NULL admis : Non</p> <p>Nom de la colonne dans la table source pour laquelle les modifications seront capturées.</p>

Tableau 75. Colonnes dans la table IBMQREP_SRC_COLS (suite)

Nom de colonne	Description
IS_KEY	<p>Type de données : SMALLINT ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si la colonne fait partie de la clé à utiliser pour la réplication ou la publication. Tous les ensembles de colonnes qui sont uniques sur la source peuvent être utilisés.</p> <p>0 (par défaut) La colonne ne fait pas partie de la clé unique. Son ordre dans le message de transaction sera le même que son ordre dans la table source.</p> <p><i>n</i> La colonne fait partie de la clé unique. Dans une clé à plusieurs colonnes, l'ordre de la colonne dans le message de transaction sera codé selon le nombre <i>n</i> que vous avez spécifié.</p> <p>Au moins une des colonnes de la table source doit avoir une valeur supérieure à 0 dans la table IBMQREP_SRC_COLS, sinon l'abonnement Q ou la publication ne sera pas valide.</p> <p>Restriction : Les colonnes d'objets LOB (Large object) et les colonnes LONG ne peuvent pas être utilisées dans la clé de publication ou de réplication.</p>
COL_OPTIONS_FLAG	<p>Type de données : CHAR(10) ; NULL admis : non, avec valeur par défaut</p> <p>Le premier caractère indique si la valeur d'image avant de la colonne est publiée ou répliquée. Le premier caractère peut avoir les valeurs suivantes :</p> <p>N Aucune image avant envoyée pour la colonne.</p> <p>D Les valeurs d'image avant sont envoyées pour les opérations de suppression.</p> <p>U Les valeurs d'image avant sont envoyées pour les opérations de mise à jour.</p> <p>Y Les valeurs d'image avant sont envoyées pour les opérations de suppression et de mise à jour.</p> <p>Par défaut : NNNNNNNNNN</p>

Table IBMQREP_SRCH_COND

IBMQREP_SRCH_COND est une table interne utilisée par le programme Q Capture pour évaluer la condition de recherche pour un abonnement Q ou une publication.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 76, à la page 522 fournit une brève description de la colonne dans la table IBMQREP_SRCH_COND.

Tableau 76. Colonne dans la table IBMQREP_SRCH_COND

Nom de colonne	Description
ASNQREQD	Type de données : CHAR(8) ; NULL admis : oui Cette colonne ne contient aucune donnée.

Table IBMQREP_SUBS

La table IBMQREP_SUBS contient des informations sur les abonnements Q ou les publications, notamment le type d'abonnement ou de publication, les conditions de recherche, les options d'envoi de données, les options de chargement et l'état de l'abonnement ou de la publication.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Clé primaire : SUBNAME

Index créé sans clause UNIQUE : SUB_ID

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 77 fournit une brève description des colonnes dans la table IBMQREP_SUBS.

Tableau 77. Colonnes dans la table IBMQREP_SUBS

Nom de colonne	Description
SUBNAME	Type de données : VARCHAR(132) ; NULL admis : non Nom de l'abonnement Q ou de la publication. Pour chaque instance du programme Q Capture, tous les noms d'abonnement Q ou de publication doivent être uniques.
SOURCE_OWNER	Type de données : VARCHAR(128) ; Null admis : non Nom de schéma ou qualificatif de haut niveau de la table source pour cet abonnement Q ou cette publication.
SOURCE_NAME	Type de données : VARCHAR(128); VARCHAR(18) pour DB2 for z/OS, version 7 et version 8, mode compatibilité ; NULL admis : non Nom de la table source pour cet abonnement Q ou cette publication.
TARGET_SERVER	Type de données : VARCHAR(18) ; NULL admis : oui Nom de la base de données ou du sous-système où les exécutions et les cibles du programme Q Apply sont définies. Pour z/OS, il s'agit d'un nom d'emplacement.
TARGET_ALIAS	Type de données : VARCHAR(8); Null admis : Oui L'Pseudonymes de la base de données DB2 qui correspond au serveur Q Apply nommé dans la colonne TARGET_SERVER.
TARGET_OWNER	Type de données : VARCHAR(128) ; NULL admis : Oui Nom de schéma ou le qualificatif de haut niveau de la table source ou de la procédure stockée pour un abonnement Q.

Tableau 77. Colonnes dans la table IBMQREP_SUBS (suite)

Nom de colonne	Description										
TARGET_NAME	Type de données : VARCHAR(128) ; NULL admis : Oui Nom de la table cible pour un abonnement Q.										
TARGET_TYPE	Type de données : INTEGER ; NULL admis : oui Indicateur signalant le type de cible de réplication. <table> <tr> <td>1</td> <td>Table utilisateur</td> </tr> <tr> <td>2</td> <td>Table de modification cohérente des données (CCD)</td> </tr> <tr> <td>3</td> <td>Réservé pour un usage ultérieur.</td> </tr> <tr> <td>4</td> <td>Réservé pour un usage ultérieur.</td> </tr> <tr> <td>5</td> <td>Procédure stockée</td> </tr> </table>	1	Table utilisateur	2	Table de modification cohérente des données (CCD)	3	Réservé pour un usage ultérieur.	4	Réservé pour un usage ultérieur.	5	Procédure stockée
1	Table utilisateur										
2	Table de modification cohérente des données (CCD)										
3	Réservé pour un usage ultérieur.										
4	Réservé pour un usage ultérieur.										
5	Procédure stockée										
APPLY_SCHEMA	Type de données : VARCHAR(128) ; NULL admis : Oui Le schéma du programme Q Apply qui applique des transactions pour cet abonnement Q.										
SENDQ	Type de données : VARCHAR(48) ; NULL admis : non Nom de la file d'attente WebSphere MQ que le programme Q Capture utilise pour envoyer des données transactionnelles pour cet abonnement Q ou cette publication. Chaque table source est appariée à une file d'attente d'envoi.										
SEARCH_CONDITION	Type de données : VARCHAR(2048) ; NULL admis : oui La condition de recherche utilisée pour filtrer les lignes pour l'abonnement Q ou la publication. Il doit s'agir d'une clause WHERE de l'instruction select annotée avec deux points directement devant les noms des colonnes source.										
SUB_ID	Type de données : INTEGER ; NULL admis : oui Un entier généré par le programme Q Capture et utilisé pour identifier de façon unique un abonnement Q dans le message subscription schema vers le programme Q Apply.										
SUBTYPE	Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut Indicateur signalant le type de réplication dans lequel un abonnement Q est impliqué ou s'il s'agit d'une publication. Indicateur spécifiant le type de réplication dans lequel l'abonnement Q est impliqué. U (par défaut) Réplication unidirectionnelle. Il s'agit de la valeur utilisée pour les publications. B Réplication bidirectionnelle. P Réplication entre homologues.										

Tableau 77. Colonnes dans la table IBMQREP_SUBS (suite)

Nom de colonne	Description
ALL_CHANGED_ROWS	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si le programme Q Capture envoie un message quand une ligne dans la table source est modifiée, même si aucune des colonnes faisant partie d'un abonnement n'a été modifiée :</p> <p>N (par défaut) Le programme Q Capture envoie un message uniquement si des colonnes faisant partie de l'abonnement Q sont modifiées.</p> <p>Y Si une ligne est modifiée dans la table source, le programme Q Capture envoie les colonnes de cette ligne qui font partie d'un abonnement Q, même si aucune d'entre elles n'a été modifiée.</p>
BEFORE_VALUES	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Cette colonne est obsolète pour la version 9 Fix Pack 1.</p>
CHANGED_COLS_ONLY	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si le programme Q Capture publie uniquement les colonnes faisant partie d'un abonnement Q ou d'une publication qui ont été modifiées. Ce champ s'applique uniquement aux opérations de mise à jour.</p> <p>Y (par défaut) Lorsque le programme Q Capture envoie une ligne mise à jour, il envoie uniquement les colonnes modifiées faisant partie d'un abonnement Q ou d'une publication.</p> <p>N Le programme Q Capture envoie toutes les colonnes d'une ligne faisant partie d'un abonnement Q ou d'une publication, qu'elles aient été modifiées ou non.</p>
HAS_LOADPHASE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si la table cible pour l'abonnement Q ou la publication sera chargée avec des données de la source :</p> <p>N (par défaut) La cible ne sera pas chargée.</p> <p>I Un chargement automatique. Le programme Q Apply appelle l'un des différents utilitaires, selon le LOAD_TYPE indiqué dans la table IBMQREP_TARGETS et selon la plateforme du serveur Q Apply et du serveur Q Capture.</p> <p>E Un chargement manuel. Une application autre que le programme Q Apply charge la table cible. Dans ce cas, l'utilisateur ou le Centre de réplcation insère le signal LOADDONE dans la table IBMQREP_SIGNAL, au niveau du serveur Q Capture, ou le programme Q Capture insère ce signal après avoir reçu le message load done.</p>

Tableau 77. Colonnes dans la table IBMQREP_SUBS (suite)

Nom de colonne	Description
STATE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur inséré par le programme Q Capture pour signaler l'état actuel de l'abonnement Q ou de la publication. L'état initial est nouveau et la valeur du champ STATE_INFO est initialement définie sur ASN7024I (nouvel abonnement Q ou nouvelle publication).</p> <p>N (par défaut) L'abonnement Q ou la publication est nouveau. Le programme Q Capture active automatiquement l'abonnement Q ou la publication lorsque le programme est démarré ou réinitialisé.</p> <p>I L'abonnement Q ou la publication est inactif. Le programme Q Capture a vu un signal CAPSTOP dans le journal ou une erreur est survenue alors que l'abonnement Q ou la publication était désactivé. Le programme Q Capture a cessé d'envoyer des messages pour cet abonnement Q ou cette publication mais a continué avec les autres.</p> <p>L L'abonnement Q est en cours de chargement. Le programme Q Capture a traité le signal CAPSTART et a envoyé le message subscription schema au programme Q Apply ou à une application utilisateur. Le programme Q Capture envoie des messages de transaction incluant des valeurs avant (before) pour toutes les colonnes et est en attente du signal LOADDONE.</p> <p>A L'abonnement Q ou la publication est actif. S'il y a une phase de chargement, le programme Q Capture a traité le signal LOADDONE et a envoyé un message load done received au programme Q Apply ou à une application utilisateur. Le programme Q Capture envoie des messages de données en fonction des options définies pour l'abonnement Q ou la publication.</p> <p>T Un état interne indiquant que le programme Q Capture lit un signal CAPSTART dans le journal pour cet abonnement Q entre homologues et que l'abonnement Q est en cours d'initialisation dans le groupe entre homologues.</p> <p>G Un état interne indiquant que le programme Q Capture lit un signal CAPSTOP dans le journal pour cet abonnement Q entre homologues et que l'abonnement Q est en cours de désactivation dans le groupe entre homologues.</p> <p>U Un état interne indiquant que le programme Q Capture crée cet abonnement Q et que l'abonnement Q est en attente d'activation.</p>
STATE_TIME	<p>Type de données : TIMESTAMP ; NULL admis : non, avec valeur par défaut</p> <p>Horodatage de la dernière modification de l'état de l'abonnement Q ou de la publication. Valeur par défaut : horodatage actuel</p>
STATE_INFO	<p>Type de données : CHAR(8) ; NULL admis : oui</p> <p>Le numéro du message ASN relatif à l'état de l'abonnement Q. Pour plus de détails, voir la table IBMQREP_CAPTRACE ou le journal de diagnostic Q Capture.</p>
STATE_TRANSITION	<p>Type de données : VARCHAR(256) FOR BIT DATA ; NULL admis : oui</p> <p>Une valeur interne utilisée pour stocker le demi état et les informations correspondantes.</p>

Tableau 77. Colonnes dans la table IBMQREP_SUBS (suite)

Nom de colonne	Description
SUBGROUP	<p>Type de données : VARCHAR(30); Null admis : Oui</p> <p>Nom du groupe d'homologues incluant cet abonnement Q. Cette colonne ne s'applique pas à une publication.</p>
SOURCE_NODE	<p>Type de données : SMALLINT ; NULL admis : non, avec valeur par défaut</p> <p>Un numéro d'identification pour le serveur source dans un abonnement Q entre homologues. Cette colonne ne s'applique pas à une publication. Par défaut : 0</p>
TARGET_NODE	<p>Type de données : SMALLINT ; NULL admis : non, avec valeur par défaut</p> <p>Un numéro d'identification pour le serveur cible dans un abonnement Q entre homologues. Cette colonne ne s'applique pas à une publication. Par défaut : 0</p>
GROUP_MEMBERS	<p>Type de données : CHAR(254) FOR BIT DATA ; NULL admis : oui</p> <p>Cette colonne est mise à jour par le programme Q Capture lorsque des membres rejoignent ou quittent un groupe entre homologues.</p>
OPTIONS_FLAG	<p>Type de données : CHAR(4) FOR BIT DATA ; NULL admis : non, avec valeur par défaut</p> <p>Réservé pour un usage ultérieur.</p>
SUPPRESS_DELETES	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur qui signale au programme Q Capture s'il doit envoyer des lignes supprimées de la table source.</p> <p>N (par défaut) Envoyer les lignes supprimées.</p> <p>Y Ne pas envoyer les lignes supprimées.</p>
DESCRIPTION	<p>Type de données : VARCHAR(200) ; NULL admis : oui</p> <p>Une description fournie par l'utilisateur de l'abonnement Q ou de la publication.</p>
TOPIC	<p>Type de données : VARCHAR(256) ; NULL admis : oui</p> <p>Une rubrique fournie par l'utilisateur à inclure dans l'en-tête du message conforme au JMS (MQRFH2) pour chaque message XML envoyé pour la publication.</p>
CAPTURE_LOAD	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Action effectuée par le programme Q Capture lorsque le journal de récupération indique qu'une opération de chargement utilisant l'utilitaire LOAD DB2 s'est produite sur la table source :</p> <p>W (valeur par défaut) Q Capture émet un message d'avertissement une fois le chargement terminé.</p> <p>R Q Capture émet un message d'avertissement puis arrête et démarre l'abonnement Q de la table source, ce qui entraîne le chargement de la table cible si une table cible est définie pour l'abonnement Q.</p>

Tableau 77. Colonnes dans la table IBMQREP_SUBS (suite)

Nom de colonne	Description
CHANGE_CONDITION	<p>Type de données : VARCHAR(2048) ; NULL admis : oui, avec valeur par défaut</p> <p>Un prédicat SQL qui utilise des variables d'enregistrement de journal pour filtrer les lignes qui sont répliquées ou publiées. Ce prédicat n'exige pas de clause WHERE comme celle utilisée dans la colonne SEARCH_CONDITION, mais il peut inclure les variables d'enregistrement de journal suivantes :</p> <p>\$OPERATION Opération DML. Les valeurs valides sont I (insérer), U (mettre à jour), et D (delete).</p> <p>\$AUTHID ID utilisateur d'une transaction.</p> <p>z/OS \$AUTHTOKEN Jeton d'autorisation (nom de travail) d'une transaction validée.</p> <p>z/OS \$PLANNAME Le nom de plan d'une transaction validée.</p> <p>Par exemple, le prédicat suivant indique à Q Capture de ne répliquer ou publier que les enregistrements de journal des opérations d'insertion validées par l'utilisateur HR pour l'abonnement Q concerné :</p> <p>\$OPERATION = 'I' AND \$AUTHID = 'HR'</p> <p>Si une clause WHERE est spécifiée dans la colonne SEARCH_CONDITION, la valeur de CHANGE_CONDITION et de SEARCH_CONDITION est combinée dans un seul prédicat à l'aide d'un opérateur AND. Cependant, vous pouvez laisser la colonne SEARCH_CONDITION vide et spécifier le prédicat complet qui inclut une clause WHERE dans cette colonne. La valeur dans cette colonne n'est pas utilisée lorsque le programme Q Apply charge la table cible.</p> <p>Pour plus de détails sur la spécification des filtres qui utilisent des enregistrements de journal, voir «Variables d'enregistrement de journal pour filtrer des lignes (réplication unidirectionnelle)», à la page 76.</p>
REPL_ADDCOL	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur qui signale au programme Q Capture s'il doit ajouter automatiquement les nouvelles colonnes de la table source à l'abonnement Q pour la table. Les colonnes sont également ajoutées à la table cible, si elles n'y figurent pas.</p> <p>N (par défaut) Ne pas ajouter automatiquement de nouvelles colonnes à l'abonnement Q.</p> <p>Y Lorsqu'une instruction ALTER TABLE ADD COLUMN est détectée, de nouvelles colonnes sont ajoutées automatiquement à l'abonnement Q pour la table source et à la table cible si elles n'existent pas dans cette dernière.</p>

Tableau 77. Colonnes dans la table IBMQREP_SUBS (suite)

Nom de colonne	Description
IGNSETNULL	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Linux UNIX Windows Indicateur qui signale au programme Q Capture s'il faut répliquer des opérations UPDATE qui résultent de la suppression de lignes parentes dans des tables avec relations d'intégrité référentielle (règle ON DELETE SET NULL). Pour la version 10.1 et ultérieure sous Linux, UNIX et Windows uniquement.</p> <p>N (par défaut) Lorsqu'une ligne parente est supprimée à la source, Q Capture réplique les opérations UPDATE à partir de lignes enfant dans lesquelles une ou plusieurs valeurs de colonne sont définies sur NULL.</p> <p>Y Q Capture ne réplique pas des opérations UPDATE au niveau de tables enfant qui résultent de la règle ON DELETE SET NULL. Seule la suppression de la ligne parente est répliquée. Cette option peut s'avérer utile si la règle ON DELETE SET NULL est activée au niveau de la base de données cible et qu'il n'est pas nécessaire que ces mises à jour soient répliquées.</p>
SCHEMA_SUBNAME	<p>Type de données : VARCHAR(64) ; NULL admis : oui</p> <p>Nom de l'abonnement de niveau schéma auquel cet abonnement Q de niveau table appartient. Cette colonne est renseignée par le programme de ligne de commande ASNCLP lorsqu'un abonnement de niveau schéma est créé et par le programme Q Capture lorsqu'il détecte une opération CREATE TABLE dans le schéma.</p>
SUB_CREATOR	<p>Type de données : VARCHAR(12) ; NULL admis : oui</p> <p>Identifie la manière dont cet abonnement Q a été créé. Les valeurs sont "asnclp", "replcenter" (le Centre de réplication) et "asnqcap" (le programme Q Capture). Lorsque les tables de contrôle de Q Capture sont migrées d'une version à la version 10 sous Linux, UNIX et Windows, la valeur de la colonne est NULL.</p>
CAPTURE_TRUNCATE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Réservé pour un usage ultérieur.</p>

Table IBMQREP_SUBS_PROF

La table IBMQREP_SUBS_PROF stocke les valeurs par défaut pour créer automatiquement des abonnements Q de niveau table lorsque des tables sont ajoutées à des bases de données avec des abonnements Q de niveau schéma. Lorsque vous créez un abonnement Q de niveau schéma, les outils d'administration de réplication remplissent cette table avec des options par défaut pour des configurations unidirectionnelles, bidirectionnelles ou entre homologues.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Clé primaire : SUBPROFNAME

Important : après avoir inséré un profil dans cette table, le profil ne doit pas être modifié, car les valeurs par défaut pour les abonnements Q de niveau table récemment créés doivent être cohérents avec les abonnements Q existants. Si vous voulez modifier le profil pour un abonnement Q de niveau schéma, utilisez le

programme de ligne de commande ASNCLP pour créer un profil avec un nom différent et associez-le à cet abonnement Q de niveau schéma.

tableau 78 donne une brève description des colonnes de la table IBMQREP_SUBS_PROF.

Tableau 78. Colonnes de la table IBMQREP_SUBS_PROF

Nom de colonne	Description
SUBPROFNAME	Type de données : VARCHAR(128) ; NULL admis : non Identificateur unique pour le profil d'abonnement Q de niveau table. Tout nom préfixé par "ASN" est réservé, car les profils suivants sont automatiquement utilisés pour une réplification unidirectionnelle ou bidirectionnelle : ASNUNI et ASNBIDI.
SUBNAME_PREFIX	Type de données : VARCHAR(4); Null admis : Oui L'identificateur qui est préfixé pour des noms d'abonnements Q de niveau table lorsque Q Capture crée automatiquement des abonnements Q basés sur des abonnements Q de niveau schéma. Par exemple, si le nom de la table source est T3 et la valeur de SUBNAME_PREFIX est PROD, le nom de l'abonnement Q qui doit être généré par le programme Q Capture doit être PRODT30001. (Si cette colonne est NULL, Q Capture génère des noms d'abonnement Q comprenant le nom de la table source plus 000n, où n >= 1).
SUBTYPE	
CHANGED_COLS_ONLY	
HAS_LOADPHASE	Pour plus d'informations sur ces colonnes, voir les descriptions dans «Table IBMQREP_SUBS», à la page 522.
SUPPRESS_DELETES	
CAPTURE_LOAD	
CONFLICT_RULE	
CONFLICT_ACTION	
ERROR_ACTION	Pour plus d'informations sur ces colonnes, voir les descriptions dans «Table IBMQREP_TARGETS», à la page 555.
OKSQLSTATES	
LOAD_TYPE	
MODELQ	

Table IBMQREP_TABVERSION

La table IBMQREP_TABVERSION est utilisée par les programmes Q Capture et Capture pour conserver une trace des différentes versions d'un table source. Le programme Q Capture ou Capture insère des lignes dans cette table lorsque l'enregistrement ou l'abonnement Q pour une table source est activé pour la première fois, puis chaque fois que la table source est modifiée.

Serveur : serveur Q Capture

Schéma par défaut : ASN

Index : LSN, TABLEID1, TABLEID2, VERSION

Index : SOURCE_OWNER ASC, SOURCE_NAME ASC

Index : TABLEID1 ASC, TABLEID2 ASC

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 79 fournit une brève description des colonnes de la table IBMQREP_TABVERSION.

Tableau 79. Colonnes de la table IBMQREP_TABVERSION

Nom de colonne	Description
LSN	Type de données : VARCHAR(16) FOR BIT DATA ; Null admis : Non Point du journal de récupération DB2 où le programme Q Capture ou Capture a détecté une nouvelle version de la table source.
TABLEID1	Type de données : SMALLINT ; NULL admis : non Identificateur de base de données (DBID) dans SYSIBM.SYSTABLES.
TABLEID2	Type de données : SMALLINT ; NULL admis : non Identificateur d'objet (OBID) dans SYSIBM.SYSTABLES.
VERSION	Type de données : INTEGER ; NULL admis : Non Nombre généré par le programme Q Capture ou Capture pour conserver une trace des différentes versions d'une table source.
SOURCE_OWNER	Type de données : VARCHAR(128) ; NULL admis : non Schéma ou qualificatif de haut niveau de la table source.
SOURCE_NAME	Type de données : VARCHAR(128) ; NULL admis : non Nom de la table source.

Tables de contrôle sur le serveur Q Apply

Les tables de contrôle sur le serveur Q Apply contiennent des paramètres d'exploitation Q Apply, des définitions d'abonnement Q, des statistiques de performances et d'autres métadonnées. Ces tables sont générées selon les options que vous avez définies dans les outils d'administration de la réplication.

Le tableau 80 décrit les tables de contrôle sur le serveur Q Apply.

Tableau 80. Tables de contrôle sur le serveur Q Apply

Nom de la table	Description
«Table IBMQREP_APPLYENQ», à la page 531	Permet de s'assurer qu'un seul programme Q Apply avec un schéma donné est en cours d'exécution par serveur Q Apply.
«Table IBMQREP_APPENVINFO», à la page 532	Stocke la valeur des variables de l'environnement d'exécution et autres informations que les outils d'administration de la réplication utilisent pour accéder aux programmes distants.
«Table IBMQREP_APPLYMON», à la page 533	Contient des statistiques sur les performances d'un programme Q Apply pour chaque file d'attente de réception.
«Table IBMQREP_APPLYPARMS», à la page 538	Contient les paramètres que vous avez définis pour contrôler les opérations d'un programme Q Apply.

Tableau 80. Tables de contrôle sur le serveur Q Apply (suite)

Nom de la table	Description
«Table IBMQREP_APPLYTRACE», à la page 545	Contient des messages d'information, d'avertissement et d'erreur provenant du programme Q Apply.
«Table IBMQREP_DELTOMB», à la page 546	Une table interne utilisée par le programme Q Apply pour enregistrer les suppressions conflictuelles dans la réplication entre homologues.
«Table IBMQREP_DONEMSG», à la page 547	Une table interne utilisée par le programme Q Apply pour enregistrer les messages qui ont été traités.
«IBMQREP_EXCEPTIONS, table», à la page 547	Contient des modifications de lignes qui ne peuvent pas être appliquées en raison de conflits, d'erreurs ou d'annulations.
«Table IBMQREP_RECVQUEUES», à la page 550	Identifie les files d'attente qu'un programme Q Apply utilise pour recevoir des messages de transaction et envoyer des messages de contrôle et contient des paramètres d'opération pour le programme Q Apply.
«Table IBMQREP_SAVERI», à la page 552	Une table interne que le programme Q Apply utilise pour stocker des contraintes d'intégrité référentielle déplacées alors que les cibles sont en cours de chargement.
«Table IBMQREP_SPILLEDROW», à la page 554	Une table interne que le programme Q Apply utilise pour garder une trace des lignes envoyées vers une file d'attente auxiliaire temporaire.
«Table IBMQREP_SPILLQS», à la page 554	Identifie les files d'attente auxiliaires temporaires qui contiendront les modifications des tables sources avant qu'elles ne soient appliquées aux cibles.
«Table IBMQREP_TRG_COLS», à la page 563	Contient des informations relatives au mappage entre les colonnes sources et cibles.
«Table IBMQREP_TARGETS», à la page 555	Contient des informations relatives aux tables cible ou aux procédures stockées et aux options des abonnements Q.

Table IBMQREP_APPLYENQ

La table IBMQREP_APPLYENQ permet de s'assurer que le schéma utilisé pour identifier le programme Q Apply et ses tables de contrôle est unique.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

La table IBMQREP_APPLYENQ permet de s'assurer que :

- **Linux UNIX Windows** Pour DB2 pour Linux, UNIX et Windows, un seul programme Q Apply avec un schéma donné est en cours d'exécution par base de données.
- **z/OS** Pour le non partage de données de DB2 for z/OS, un seul programme Q Apply avec un schéma donné est en cours d'exécution par sous-système.

- **z/OS** Pour le partage de données de DB2 for z/OS, un seul programme Q Apply avec un schéma donné est en cours d'exécution par groupe de partage de données.

En cours d'exécution, un programme Q Apply verrouille exclusivement cette table. Si le programme Q Apply démarre deux fois, la deuxième instance sera placée en attente sur verrouillage sur cette table. La table est créée vide.

Le tableau 81 fournit une brève description de la colonne dans la table IBMQREP_APPLYENQ.

Tableau 81. Colonne de la table IBMQREP_APPLYENQ

Nom de colonne	Description
LOCKNAME	Type de données : INTEGER ; NULL admis : oui
	Cette colonne ne contient aucune donnée.

Table IBMQREP_APPENVINFO

La table IBMQREP_APPENVINFO contient huit lignes utilisées pour stocker la valeur des variables de l'environnement d'exécution et autres informations que les outils d'administration de la réplication utilisent pour accéder aux programmes distants.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 82, à la page 533 donne une brève description des colonnes de la table IBMQREP_APPENVINFO.

Tableau 82. Colonnes de la table IBMQREP_APPENVINFO

Nom de colonne	Description
NAME	<p>Type de données : VARCHAR(30) ; NULL admis : non</p> <p>STARTTIME Horodatage lorsque le programme Q Apply a démarré.</p> <p>HOSTNAME Nom d'hôte TCP/IP du serveur lorsque le programme Q Apply est en cours d'exécution.</p> <p>LOGFILE Chemin d'accès et nom du fichier du fichier journal de diagnostic du programme Q Apply.</p> <p>TMPDIR Chemin d'accès au répertoire où est située la clé de communication interprocessus (IPC) du programme Q Apply.</p> <p>ASNUSEMQCLIENT Valeur de la variable d'environnement de réplication Q ASNUSEMQCLIENT.</p> <p>MQSERVER Valeur de la variable d'environnement WebSphere MQ MQSERVER.</p> <p>MQCHLLIB Valeur de la variable d'environnement WebSphere MQ MQCHLLIB.</p> <p>MQCHLTAB Valeur de la variable d'environnement WebSphere MQ MQCHLTAB.</p>
VALUE	<p>Type de données : VARCHAR(3800); NULL admis : Oui</p> <p>Pour chaque ligne de la table IBMQREP_APPENVINFO, la colonne VALUE contient la valeur associée à la colonne NAME correspondante.</p>

Table IBMQREP_APPLYMON

Le programme Q Apply insère périodiquement des lignes dans la table IBMQREP_APPLYMON pour enregistrer des statistiques de performance, une ligne par file d'attente de réception.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Index créé sans clause UNIQUE : MONITOR_TIME DESC

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

La valeur que vous avez définie pour MONITOR_INTERVAL dans la table IBMQREP_APPLYPARMS détermine la fréquence à laquelle le programme Q Apply insère des lignes dans la table de contrôle. La valeur MONITOR_LIMIT détermine la durée pendant laquelle les lignes restent dans la table avant d'être susceptibles d'être élaguées.

Le tableau 83, à la page 534 fournit une brève description des colonnes dans la table IBMQREP_APPLYMON.

Tableau 83. Colonnes dans la table IBMQREP_APPLYMON

Nom de colonne	Description
MONITOR_TIME	Type de données : TIMESTAMP; Null admis : Non Horodatage sur le serveur Q Apply lorsque la ligne a été insérée dans la table IBMQREP_APPLYMON.
RECVQ	Type de données : VARCHAR(48) ; NULL admis : non Nom de la file d'attente de réception à laquelle cette ligne des statistiques de performances Q Apply se rapporte.
QSTART_TIME	Type de données : TIMESTAMP; Null admis : Non Horodatage sur le serveur Q Apply lorsque la file d'attente de réception a été démarrée.
CURRENT_MEMORY	Type de données : INTEGER ; NULL admis : Non Quantité de mémoire en octets que l'unité d'exécution du navigateur Q Apply a utilisée pour lire les transactions provenant de cette file d'attente.
QDEPTH	Type de données : INTEGER ; NULL admis : Non Profondeur de la file d'attente (nombre de messages dans la file d'attente). La colonne Q_PERCENT_FULL exprime le remplissage de la file d'attente en pourcentage.
END2END_LATENCY	Type de données : INTEGER ; NULL admis : Non Temps moyen en millisecondes écoulé entre le moment de la validation des transactions sur la table source et le moment de leur validation sur la cible.
QLATENCY	Type de données : INTEGER ; NULL admis : Non Temps moyen en millisecondes écoulé entre le moment où le programme Q Capture a mis les messages sur la file d'attente d'envoi et le moment où le programme Q Apply les a obtenus de la file d'attente de réception.
APPLY_LATENCY	Type de données : INTEGER ; NULL admis : Non Temps moyen en millisecondes écoulé entre le moment où le programme Q Apply lit les transactions de la file d'attente de réception et le moment où elles sont validées sur la cible.
TRANS_APPLIED	Type de données : INTEGER ; NULL admis : Non Nombre total de transactions provenant de cette file d'attente de réception que Q Apply a validées sur la cible.
ROWS_APPLIED	Type de données : INTEGER ; NULL admis : Non Nombre total d'opérations d'insertion, de mise à jour et de suppression provenant de cette file d'attente de réception que le programme Q Apply a appliquées sur la cible.
TRANS_SERIALIZED	Type de données : INTEGER ; NULL admis : Non Nombre total de transactions qui ont été en conflit avec une autre transaction (soit en raison d'un conflit de lignes ou d'un conflit d'intégrité référentielle). Dans ce cas, le programme Q Apply suspend le traitement parallèle et applique les modifications de lignes dans la transaction afin qu'elles soient validées sur la source.
RI_DEPENDENCIES	Type de données : INTEGER ; NULL admis : Non Nombre total de conflits d'intégrité référentielle qui ont été détectés, obligeant à mettre les transactions en série.

Tableau 83. Colonnes dans la table IBMQREP_APPLYMON (suite)

Nom de colonne	Description
RI_RETRIES	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Nombre de fois où le programme Q Apply a dû appliquer une nouvelle fois les modifications de lignes en raison de conflits d'intégrité référentielle apparus lorsque les transactions dont elles faisaient partie ont été exécutées en parallèle.</p>
DEADLOCK_RETRIES	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Nombre de fois où le programme Q Apply a appliqué à nouveau des modifications de lignes en raison de dépassements du délai d'attente de verrouillage et de blocages.</p>
ROWS_NOT_APPLIED	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Nombre de lignes qui n'ont pas pu être appliquées et on été entrées dans la table IBMQREP_EXCEPTIONS.</p>
MONSTER_TRANS	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Nombre de transactions qui ont dépassé MEMORY_LIMIT pour la file d'attente de réception paramétré dans la table IBMQREP_RECVQUEUES.</p>
MEM_FULL_TIME	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Nombre de secondes pendant lesquelles le programme Q Apply n'a pas pu générer des transactions à partir de cette file d'attente de réception car ses agents utilisaient toute la mémoire disponible pour appliquer les transactions.</p>
APPLY_SLEEP_TIME	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Nombre de millisecondes d'inactivation des agents Q Apply de cette file d'attente en attendant un travail.</p>
SPIILLED_ROWS	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Nombre de lignes que le programme Q Apply a envoyées aux files d'attente auxiliaires temporaires pendant que les cibles étaient en cours de chargement ou pendant que les abonnements Q étaient placés dans un état de déversement par le paramètre spillsub de la commande MODIFY ou asnqcmd.</p>
SPIILLEDROWSAPPLIED	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Nombre de lignes auxiliaires qui ont été appliquées à la cible.</p>
OLDEST_TRANS	<p>Type de données : TIMESTAMP; Null admis : Non</p> <p>Horodatage basé sur l'heure local du serveur Q Apply permettant de déterminer l'état de rattrapage de Q Apply par rapport à la source. A chaque intervalle de surveillance, OLDEST_TRANS représente :</p> <ul style="list-style-type: none"> • Si Q Apply traite des transactions, l'heure de validation de la source pour laquelle toutes les transactions ont été appliquées à la cible à ce stade (D'autres transactions plus récentes ont aussi pu être validées. Le programme Q Apply traitant les transactions en parallèle, les heures de validation de ces transactions plus récentes ne se rapportent pas à un point auquel toutes les transactions précédentes ont été appliquées). • La dernière heure de pulsation, si aucune transaction n'est en cours de traitement et que le message de pulsation est arrivé après la transaction appliquée la plus ancienne. • La valeur 1900-01-01-00.00.000000 si le programme Q Apply n'a vu aucun message (transaction ou pulsation).

Tableau 83. Colonnes dans la table IBMQREP_APPLYMON (suite)

Nom de colonne	Description
OLDEST_INFLT_TRANS	<p>Type de données : TIMESTAMP; Null admis : Non</p> <p>A chaque intervalle de surveillance, OLDEST_INFLT_TRANS représente :</p> <ul style="list-style-type: none"> • Si Q Apply traite des transactions, l'heure de validation de la source de la transaction actuellement en cours la plus ancienne. Une transaction en cours n'a pas été complètement appliquée et validée sur la cible. • La valeur NULL si Q Apply ne traite pas de transaction. • La valeur NULL si Q Apply n'a pas vu de message de transaction jusqu'à présent. <p>OLDEST_INFLT_TRANS ne prend pas en compte les messages de pulsation. Il ne prend en compte que les messages de transaction. De plus, cette valeur ne reflète pas l'état de rattrapage de Q Apply par rapport à la source car l'heure de validation de la source appartient à une transaction que n'a pas encore été complètement traitée ou validée.</p>
OKSQLSTATE_ERRORS	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Nombre de modifications de lignes qui ont entraîné une erreur SQL définie comme acceptable dans le champ OKSQLSTATES de la table IBMQREP_TARGETS. Le programme Q Apply ignore ces erreurs.</p>
HEARTBEAT_LATENCY	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Temps moyen en millisecondes écoulé entre le moment où les messages de pulsation ont été envoyés par le programme Q Capture et le moment où ils ont été reçus par le programme Q Apply.</p>
KEY_DEPENDENCIES	<p>Type de données : INTEGER ; NULL admis : Non</p> <p>Nombre total de contraintes de clé de réplication qui ont été détectées, obligeant à mettre les transactions en série.</p>
UNIQ_DEPENDENCIES	<p>Type de données : INTEGER; NULL admis : Non</p> <p>Nombre total de contraintes d'index à entrées uniques qui ont été détectées, obligeant à mettre les transactions en série.</p>
UNIQ_RETRIES	<p>Type de données : INTEGER; NULL admis : Non</p> <p>Nombre de fois où le programme Q Apply a essayé d'appliquer à nouveau des lignes qui n'ont pas été appliquées en parallèle en raison de contraintes d'index à entrées uniques.</p>
JOB_DEPENDENCIES	<p>Type de données : INTEGER; NULL admis : Non</p> <p>Nombre de transactions différées en raison de dépendances d'ID de corrélation.</p>
CAPTURE_LATENCY	<p>Type de données : INTEGER; NULL admis : Non</p> <p>Temps moyen en millisecondes écoulé entre le moment de la validation des transactions sur la table source et le moment où Q Capture met dans la file d'attente d'envoi le dernier message pour les transactions.</p>

Tableau 83. Colonnes dans la table IBMQREP_APPLYMON (suite)

Nom de colonne	Description
OLDEST_COMMIT_LSN	<p>Type de données : VARCHAR(16) FOR BIT DATA ; NULL admis : Yes</p> <p>Le numéro d'ordre du journal de validation provenant du journal de récupération source et correspondant à la plus ancienne transaction appliquée. Toutes les transactions avec des numéros de séquence inférieurs ont été appliquées. Certaines transactions plus récentes sont également susceptibles d'être appliquées. Si aucune transaction source n'est encore validée, la valeur de cette colonne ne contient que des zéros, par exemple : 'x'00000000000000000000'.</p> <p>z/OS Vous pouvez utiliser la valeur OLDEST_COMMIT_LSN sous z/OS comme valeur du paramètre maxcmtseq lorsque vous devez redémarrer le programme Q Capture à partir d'un emplacement connu dans le journal de récupération.</p>
ROWS_PROCESSED	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Nombre de lignes lues depuis les files d'attente de réception et appliquées mais pas encore validées sur la cible.</p>
Q_PERCENT_FULL	<p>Type de données : SMALLINT ; NULL admis : oui, avec valeur par défaut</p> <p>Remplissage de la file d'attente de réception exprimé en pourcentage où l'attribut MAXDEPTH de la file d'attente est 100 %. La colonne QUEUE_DEPTH exprime le remplissage sous la forme d'un nombre de messages.</p>
OLDEST_COMMIT_SEQ	<p>Linux UNIX Windows Type de données : VARCHAR(16) FOR BIT DATA ; NULL admis : Yes</p> <p>Marqueur de journal interne qui représente la dernière transaction appliquée par le programme Q Apply avant que toutes les transactions précédentes aient également été appliquées. La valeur est un horodatage formaté avec une précision de l'ordre de la nanoseconde qui est codée sous la forme de deux entiers, des secondes et desnanosecondes dans une séquence à 10 octets.</p> <p>Vous pouvez utiliser la valeur OLDEST_COMMIT_SEQ sous Linux, UNIX et Windows comme valeur du paramètre maxcmtseq lorsque vous devez redémarrer le programme Q Capture à partir d'un emplacement connu dans le journal de récupération.</p>
MQ_BYTES	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Nombre d'octets des données lues à partir de toutes les files d'attente de réception dans l'intervalle de surveillance, y compris les données de message et l'en-tête de message.</p>
MQGET_TIME	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Nombre de millisecondes dans l'intervalle de surveillance que le programme Q Apply a passé en interagissant avec l'API de WebSphere MQ pour récupérer des messages à partir de toutes les files d'attente de réception. La valeur inclut des appels MQGET infructueux.</p>
NUM_MQGETS	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Le nombre de fois que le programme Q Apply a utilisé avec succès l'appel MQGET pour récupérer un message à partir de toutes les files d'attente de réception dans l'intervalle de surveillance. Les appels MQGET qui échouent dans la récupération d'un message à partir de la file d'attente de réception ne sont pas comptés.</p>

Table IBMQREP_APPLYPARMS

La table IBMQREP_APPLYPARMS contient des paramètres que vous pouvez modifier pour contrôler l'opération du programme Q Apply. Par exemple, vous pouvez définir le nom du gestionnaire de files d'attente que le programme Q Apply utilisera ou combien de temps le programme Q Apply conservera les données dans la table IBMQREP_APPLYMON avant de les élaguer. Le programme Q Apply lit les modifications de cette table uniquement au démarrage.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Index à entrées uniques : QMGR

Cette table contient des informations que vous pouvez mettre à jour à l'aide de SQL.

La table IBMQREP_APPLYPARMS contient une seule ligne. Si cette table n'a pas de ligne ou plusieurs lignes, le programme Q Apply ne s'exécutera pas.

Le tableau 84 fournit une brève description des colonnes dans la table IBMQREP_APPLYPARMS.

Tableau 84. Colonnes dans la table IBMQREP_APPLYPARMS

Nom de colonne	Description
QMGR	Type de données : VARCHAR(48) ; NULL admis : non Nom du gestionnaire de files d'attente WebSphere MQ que le programme Q Apply utilise.
MONITOR_LIMIT	Type de données : INTEGER ; NULL admis : non, avec valeur par défaut Nombre de minutes pendant lesquelles les lignes restent dans la table IBMQREP_APPLYMON avant d'être susceptibles d'être élaguées. A chaque intervalle d'élagage, des lignes de la table IBMQREP_APPLYMON sont élaguées si elles sont plus anciennes que cette limite en fonction de l'horodatage actuel. Par défaut : 10080
TRACE_LIMIT	Type de données : INTEGER ; NULL admis : non, avec valeur par défaut Nombre de minutes pendant lesquelles les lignes restent dans la table IBMQREP_APPLYTRACE avant d'être susceptibles d'être élaguées. A chaque intervalle d'élagage, des lignes de la table IBMQREP_APPLYTRACE sont élaguées si elles sont plus anciennes que cette limite en fonction de l'horodatage actuel. Par défaut : 10080
MONITOR_INTERVAL	Type de données : INTEGER ; NULL admis : non, avec valeur par défaut Fréquence, en millisecondes, à laquelle le programme Q Apply ajoute une ligne à la table IBMQREP_APPLYMON. Valeurs par défaut : 60000 millisecondes (1 minute) sous z/OS ; 30000 millisecondes (30 secondes) sous Linux, UNIX, et Windows
PRUNE_INTERVAL	Type de données : INTEGER ; NULL admis : non, avec valeur par défaut Fréquence, en secondes, à laquelle le programme Q Apply élague automatiquement des lignes dans les tables IBMQREP_APPLYMON et IBMQREP_APPLYTRACE. Par défaut : 300

Tableau 84. Colonnes dans la table IBMQREP_APPLYPARMS (suite)

Nom de colonne	Description
AUTOSTOP	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant au programme Q Apply s'il doit s'arrêter lorsque toutes les files d'attente de réception ont été vidées une première fois.</p> <p>N (par défaut) Le programme Q Apply continue son exécution après que toutes les files d'attente de réception ont été vidées une première fois.</p> <p>Y Le programme Q Apply s'arrête après que toutes les files d'attente de réception ont été vidées une première fois.</p>
LOGREUSE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si le programme Q Apply réutilise le fichier journal Q Apply ou fait des ajouts à ce journal.</p> <p>N (par défaut) Le programme Q Apply ajoute de nouvelles informations à un fichier journal Q Apply existant lorsqu'il redémarre.</p> <p>Y Au redémarrage, le programme Q Apply réutilise le fichier journal en nettoyant le fichier puis en écrivant dans le fichier vierge.</p>
LOGSTDOUT	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si le programme Q Apply envoie des messages de journal aux sorties autres que son fichier journal.</p> <p>N (par défaut) Le programme Q Apply dirige la plupart des messages de journal vers le fichier journal uniquement.</p> <p>Y Le programme Q Apply envoie des messages de journal au fichier journal et à la console (stdout).</p> <p>Les messages d'activation et de désactivation de l'initialisation, de l'arrêt et de l'abonnement vont vers la console (stdout) et vers le fichier journal, quelle que soit la valeur de ce paramètre.</p>
APPLY_PATH	<p>Type de données : VARCHAR(1040) ; NULL admis : oui, avec valeur par défaut</p> <p>Chemin où les fichiers créés par le programme Q Apply sont stockés. Il s'agit, par défaut, du répertoire dans lequel le programme Q Apply est démarré.</p>
ARCH_LEVEL	<p>Type de données : CHAR(4) ; NULL admis : non, avec valeur par défaut</p> <p>Version des tables de contrôle. Pour Replication Server Version 10 sous z/OS la valeur est 100Z. Les autres valeurs de ARCH_LEVEL sont 0802, 0901, 0905, 0907 et 0973 pour Replication Server version 9.7 avec groupe de correctifs sous Linux, UNIX et Windows.</p> <p>Avertissement : Lors de la mise à jour de la table IBMQREP_APPLYPARMS, ne modifiez pas la valeur de cette colonne.</p>

Tableau 84. Colonnes dans la table IBMQREP_APPLYPARMS (suite)

Nom de colonne	Description
TERM	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant que le programme Q Apply s'arrête si la cible DB2 ou le gestionnaire de files d'attente est indisponible.</p> <p>Y (par défaut) Le programme Q Apply s'arrête si DB2 ou le gestionnaire de files d'attente est indisponible.</p> <p>N Le programme Q Apply continue de s'exécuter si DB2 ou le gestionnaire de files d'attente est indisponible. Lorsque DB2 ou le gestionnaire de files d'attente est disponible, Q Apply commence à appliquer les transactions là où il s'est arrêté sans qu'il soit nécessaire de redémarrer le programme.</p>
PWDFILE	<p>Type de données : VARCHAR(48) ; NULL admis : oui, avec valeur par défaut</p> <p>Nom du fichier de mot de passe codé que le programme Q Apply utilise pour se connecter au programme Q Capture si l'abonnement Q appelle une charge interne de la cible. La commande asnpwd crée ce fichier par défaut dans le répertoire spécifié dans la colonne APPLY_PATH.</p>
DEADLOCK_RETRIES	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Nombre de fois que le programme Q Apply tente d'appliquer à nouveau les modifications sur les tables cible ou d'effectuer des insertions dans ses tables de contrôle à la suite de blocages SQL. Si le blocage se produit sur une table cible, le programme Q Apply continue d'essayer jusqu'à atteindre la limite que vous avez définie. Une fois la limite atteinte, si les blocages persistent, l'unité d'exécution du navigateur s'arrête. Par défaut : 3 tentatives.</p>
SQL_CAP_SCHEMA	<p>Type de données : VARCHAR(128) ; NULL admis : oui, avec valeur par défaut</p> <p>Schéma des tables de contrôle de capture qu'utilise le programme Q Apply pour gérer les tables cible de modification cohérente des données enregistrées comme sources de réplication SQL. Cette colonne contient une valeur pour que le programme Q Apply puisse gérer les configurations de distribution (ventilation) des données.</p>
LOADCOPY_PATH	<p>Type de données : VARCHAR(1040) ; NULL admis : oui, avec valeur par défaut</p> <p>Spécifie le chemin d'accès où l'utilitaire DB2 LOAD crée une copie des données chargées sur le serveur principal pour une configuration qui implique l'utilitaire DB2 HADR (High Availability Disaster Recovery). Lorsque vous définissez ce paramètre, le programme Q Apply démarre l'utilitaire LOAD avec l'option permettant de créer la copie lorsque Q Apply charge la table cible. Le serveur secondaire de la configuration HADR recherche alors les données copiées dans ce chemin.</p>
NICKNAME_COMMIT_CT	<p>Type de données : INTEGER ; NULL admis : oui, avec valeur par défaut</p> <p>Spécifie le nombre de lignes après lequel l'utilitaire IMPORT de DB2 valide les modifications des pseudonymes qui font référence à la table cible pendant le processus de chargement. Ce paramètre s'applique uniquement aux chargements automatiques pour les cibles fédérées, qui emploient l'utilitaire IMPORT.</p> <p>La valeur par défaut est nickname_commit_ct=10.</p> <p>Ce paramètre peut être utilisé pour ajuster les performances de l'utilitaire IMPORT de DB2 en réduisant le nombre de validations pour les cibles fédérées.</p>

Tableau 84. Colonnes dans la table IBMQREP_APPLYPARMS (suite)

Nom de colonne	Description
SPILL_COMMIT_COUNT	<p>Type de données : INTEGER ; NULL admis : oui, avec valeur par défaut</p> <p>Indique le nombre de lignes regroupées dans une portée de validation par les agents de déversement Q Apply qui appliquent les données répliquées dans le cadre de l'opération de chargement. La valeur par défaut est spill_commit_count=10. En augmentant le nombre de lignes appliquées avant l'exécution d'une commande COMMIT, vous pouvez améliorer les performances en limitant les ressources E/S nécessaires pour des validations fréquentes. Il convient de trouver un compromis entre l'amélioration des performances et les risques de conflit d'accès à la table cible et à la table de contrôle IBMQREP_SPILLEDROW.</p>
LOAD_DATA_BUFF_SIZE	<p>Type de données : INTEGER ; NULL admis : oui, avec valeur par défaut</p> <p>Indique le nombre de pages de 4 Ko que l'utilitaire LOAD de DB2 LOAD doit utiliser comme espace de mémoire tampon pour le transfert des données au sein de l'utilitaire lors du chargement initial de la table cible. Ce paramètre s'applique uniquement aux chargements automatiques à l'aide de l'utilitaire DB2 LOAD.</p> <p>Par défaut, le programme Q Apply démarre l'utilitaire en indiquant d'utiliser une mémoire tampon de 8 pages. En indiquant un nombre de pages plus élevé, il est possible d'améliorer de manière significative les performances de chargement des tables MDC qui sont des cibles de réplication.</p>
CLASSIC_LOAD_FILE_SIZE	<p>Type de données : INTEGER ; NULL admis : oui avec valeur par défaut</p> <p>z/OS Indique le nombre approximatif de lignes dans les tables ou les vues d'une source de données de réplication Classic, à charger dans les tables cible. Le programme Q Apply utilise cette estimation pour calculer l'allocation DASD pour le fichier, utilisée comme entrée pour l'utilitaire de chargement. La valeur par défaut est de 500 000 lignes. Utilisez ce paramètre si l'allocation par défaut n'est pas suffisante.</p> <p>Ce paramètre s'applique uniquement aux chargements automatiques pour les tables cible z/OS à partir de sources Classic.</p>
MAX_PARALLEL_LOADS	<p>Type de données : INTEGER ; NULL admis : oui, avec valeur par défaut</p> <p>Indique le nombre maximal d'opérations de chargement automatiques des tables cible que le programme Q Apply peut démarrer en même temps pour une file d'attente de réception donnée. La valeur par défaut dépend de la plateforme du serveur cible :</p> <p>z/OS</p> <p>Sous z/OS la valeur par défaut correspond à un chargement à la fois en raison des incidents éventuels provoqués par la procédure stockée DSNUTILS que le programme Q Apply utilise pour appeler l'utilitaire DB2 LOAD. En fonction de votre environnement, vous pouvez essayer de définir une valeur supérieure à max_parallel_loads=1. Si des erreurs se produisent, réinitialisez la valeur sur 1.</p> <p>Linux UNIX Windows</p> <p>Sous Linux, UNIX et Windows, la valeur par défaut est 15 chargements parallèles.</p>

Tableau 84. Colonnes dans la table IBMQREP_APPLYPARMS (suite)

Nom de colonne	Description
COMMIT_COUNT	<p>Type de données : INTEGER ; NULL admis : oui, avec valeur par défaut</p> <p>Indique le nombre de transactions que chaque unité d'exécution agent Q Apply applique à la table cible dans une portée de validation. Par défaut (commit_count=1), les unités d'exécution agent valident après chaque transaction qu'ils appliquent. En augmentant la valeur du paramètre commit_count et en regroupant davantage de transactions dans la portée de validation, vous pouvez améliorer les performances.</p> <p>Recommandation : N'utilisez une valeur plus élevée pour commit_count que dans le cas d'un verrouillage au niveau de la ligne. Définissez ce paramètre avec soin lorsqu'il est utilisé avec un grand nombre d'unités d'exécution agent, il peut générer une escalade de verrous et provoquer des dépassements du délai d'attente des verrous et des blocages de relances.</p>
INSERT_BIDI_SIGNAL	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Si les programmes Q Capture et Q Apply utilisent des insertions de signaux P2PNORECAPTURE pour prévenir la recapture de transactions dans une réplification bidirectionnelle.</p> <p>Y (par défaut) Le programme Q Apply insère des signaux P2PNORECAPTURE dans la table IBMQREP_SIGNAL pour donner l'instruction au programme Q Capture situé sur le même serveur de ne pas recapturer les transactions validées sur ce serveur.</p> <p>N Le programme Q Apply n'insère pas de signaux P2PNORECAPTURE. En revanche, vous insérez les informations AUTHTKN de Q Apply dans la table IBMQREP_IGNTRAN, ce qui donne l'instruction au programme Q Capture situé sur le même serveur de ne pas capturer les transactions issues du programme Q Apply, sauf pour les insertions dans la table IBMQREP_SIGNAL.</p>
APPLY_ALIAS	<p>Type de données : VARCHAR(8); Null admis : Oui</p> <p>Le nom d'alias pour la base de données ou le sous-système qui est utilisé comme serveur Q Apply. Il s'agit de l'alias tel que catalogué sur le système sur lequel s'exécutent les outils d'administration de la réplification, lesquels sont également utilisés pour se connecter à la base de données cible ou au sous-système pour créer des tables de contrôle Q Apply. Cette colonne est renseignée par le Centre de réplification ou le programme de ligne de commande ASNCLP lors de la création des tables de contrôle.</p>
STARTALLQ	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur qui signale au programme Q Apply s'il doit activer toutes les files d'attente de réception qui ne sont pas déjà actives lorsque ce programme démarre. Les files d'attente de réception qui sont déjà actives sont toujours traitées lorsque Q Apply démarre.</p> <p>Y Lorsque le programme Q Apply démarre, il active toutes les files d'attente de réception qui ne sont pas déjà actives (A).</p> <p>N Lorsque le programme Q Apply démarre, il n'active pas toutes les files d'attente de réception qui sont inactives (I).</p> <p>La valeur par défaut de cette colonne est N pour z/OS et Y pour Linux, UNIX et Windows.</p>

Tableau 84. Colonnes dans la table IBMQREP_APPLYPARMS (suite)

Nom de colonne	Description
PRUNE_BATCH_SIZE	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Le nombre de lignes qui sont supprimées de la table IBMQREP_DONEMSG dans une portée de validation lorsque la valeur de PRUNE_METHOD est 2. La valeur par défaut est 1000 lignes. La valeur minimale est 2 lignes.</p>
PRUNE_METHOD	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>z/OS Méthode que le programme Q Apply emploie pour supprimer les lignes inutiles de la table IBMQREP_DONEMSG.</p> <p>1 Q Apply supprime un message de la file d'attente de réception, interroge la table IBMQREP_DONESG pour vérifier si des données provenant d'un message ont été appliquées, puis élague la ligne correspondante dans IBMQREP_DONEMSG en exécutant une instruction SQL individuelle.</p> <p>2 (valeur par défaut) Q Apply élague des groupes de lignes basées sur la valeur PRUNE_BATCH_SIZE. Une unité d'exécution d'élagage distincte enregistre les messages qui ont été appliqués, puis produit une table DELETE basée sur une plage de valeurs par rapport à la table IBMQREP_DONEMSG.</p>
IGNBADDATA	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Remarque : Cette colonne s'applique uniquement si le programme Q Apply utilise les Composants internationaux pour l'Unicode (ICU) pour la conversion de page de code (si la page de code de la base de données source et la page de code utilisée par Q Apply sont différentes). Si le programme Q Apply vérifie les caractères non autorisés dans les données de la source et continue le traitement s'il trouve des caractères non autorisés.</p> <p>Y Q Apply recherche les caractères non autorisés.</p> <p>N (par défaut) Q Apply ne vérifie pas les caractères non autorisés et ne signale pas les exceptions pour les caractères non autorisés.</p>
RICHLVL	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Niveau de vérification de l'intégrité référentielle. Par défaut, le programme Q Apply vérifie les dépendances basées sur RI entre les transactions, afin de s'assurer que les lignes dépendantes sont appliquées dans le bon ordre.</p> <p>0 Q Apply ne vérifie pas les dépendances basées sur RI.</p> <p>2 (valeur par défaut) Q Apply vérifie les dépendances basées sur RI lorsqu'une valeur clé dans la table parente ou une ligne est supprimée de la table parente.</p> <p>5 Q Apply vérifie les dépendances basées sur RI lorsqu'une valeur clé est mise à jour dans la table parente ou une ligne mise à jour dans la table parente, ou une ligne est supprimée de la table parente.</p>

Tableau 84. Colonnes dans la table IBMQREP_APPLYPARMS (suite)

Nom de colonne	Description
TRACE_DDL	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Lors de la réplication d'opération DDL au niveau de la base de données source, cette colonne indique si le texte SQL de l'opération exécuté par le programme Q Apply au niveau de la base de données cible est consigné.</p> <p>N (par défaut) Les opérations DDL répliquées ne sont pas consignées.</p> <p>Y Le programme Q Apply émet un message ASN dans son fichier journal et émet une sortie standard ainsi qu'une table IBMQREP_APPLYTRACE avec le texte de l'instruction SQL qui a été utilisée pour répliquer le DDL source. Le texte SQL est tronqué à 1024 caractères.</p>
REPORT_EXCEPTIONS	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur qui détermine si le programme Q Apply insère des données dans la table IBMQREP_EXCEPTIONS en cas de conflit ou d'erreur SQL au niveau de la table cible, sauf que la ligne est de toute façon appliquée car l'action liée au conflit et définie pour l'abonnement Q était F (forcer).</p> <p>Y (par défaut) Q Apply insère des données dans la table IBMQREP_EXCEPTIONS, indépendamment de l'application de la ligne à l'origine de l'exception.</p> <p>N Q Apply n'insère pas de données dans la table IBMQREP_EXCEPTIONS lorsque la ligne à l'origine d'une exception est appliquée ; les données ne sont insérées que si la ligne n'est pas appliquée. Avec ce réglage, Q Apply tolère également des erreurs de conversion de page de code lors de l'écriture d'un texte SQL dans la table IBMQREP_EXCEPTIONS et poursuit le traitement normal.</p>
ORACLE_EMPTY_STR	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur qui précise si le programme Q Apply remplace une chaîne vide par un espace dans les colonnes VARCHAR. DB2 autorise les chaînes vides dans les colonnes VARCHAR. Lorsqu'une colonne VARCHAR DB2 source est mappée sur une cible Oracle, ou sur un serveur DB2 exécuté avec le mode compatibilité Oracle, la chaîne vide est convertie en valeur NULL. L'opération échoue lorsque la colonne cible est définie avec une sémantique NOT NULL.</p> <p>Avec oracle_empty_str=y, Q Apply remplace la valeur NULL par un espace juste avant d'appliquer les données à la cible et après toute conversion de page de code. Si vous utilisez des expressions SQL dans des abonnements Q, prenez en compte les remarques suivantes :</p> <p>Y Q Apply remplace la valeur NULL par un espace d'un caractère juste avant d'appliquer les données à la cible et toute conversion de page de code.</p> <p>N (par défaut) Q Apply applique la valeur NULL même si cette action peut se traduire par une erreur, si la colonne cible n'accepte pas de valeurs NULL.</p>

Tableau 84. Colonnes dans la table IBMQREP_APPLYPARMS (suite)

Nom de colonne	Description
LOGMARKERTZ	<p>Type de données : CHAR(8) ; NULL admis : non, avec valeur par défaut</p> <p>Valeur qui détermine le fuseau horaire utilisé par le programme Q Apply lorsqu'il insère des données de validation de source dans la colonne IBMSNAP_LOGMARKER des tables de modification cohérente des données (CCD) ou des tables de point de cohérence (PIT).</p> <p>Par défaut : GMT (temps moyen de Greenwich) Q Apply enregistre l'horodatage de validation de source en utilisant le temps moyen de Greenwich (GMT).</p> <p>LOCAL Q Apply insère l'horodatage de validation de source en utilisant l'heure locale du serveur Q Apply.</p> <p>Remarque : Comme la valeur de la colonne IBMSNAP_LOGMARKER enregistre l'heure à laquelle une ligne a été validée dans la base de données source, le paramètre Logmarkertz=local n'est utile que lorsque les serveurs Q Capture et Q Apply sont situés dans le même fuseau horaire avec une heure d'été (ou autre) identique.</p>

Table IBMQREP_APPLYTRACE

La table IBMQREP_APPLYTRACE contient des messages d'information, d'avertissement et d'erreur provenant du programme Q Apply. Vous pouvez configurer un élagage automatique de cette table à l'aide du paramètre TRACE_LIMIT dans la table IBMQREP_APPLYPARMS.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Index créé sans clause UNIQUE : TRACE_TIME

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 85 fournit une brève description des colonnes dans la table IBMQREP_APPLYTRACE.

Tableau 85. Colonnes dans la table IBMQREP_APPLYTRACE

Nom de colonne	Description
OPERATION	<p>Type de données : CHAR(8) ; NULL admis : non</p> <p>Type de message provenant du programme Q Apply :</p> <p>INFOS Décrit les actions prises par le programme Q Apply.</p> <p>AVERTISSEMENT Décrit les conditions qui peuvent entraîner des erreurs pour le programme Q Apply.</p> <p>ERREUR Décrit les erreurs rencontrées par le programme Q Apply.</p>
TRACE_TIME	<p>Type de données : TIMESTAMP; Null admis : Non</p> <p>Heure sur le serveur Q Apply à laquelle la ligne a été insérée dans cette table.</p>

Tableau 85. Colonnes dans la table IBMQREP_APPLYTRACE (suite)

Nom de colonne	Description
DESCRIPTION	Type de données : VARCHAR(1024) ; NULL admis : non Identificateur de message ASN suivi par le texte de message. Cette colonne contient un texte en anglais uniquement.
REASON_CODE	Type de données : INTEGER ; NULL admis : oui Code anomalie pour le message d'erreur de réplication.
MQ_CODE	Type de données : INTEGER ; NULL admis : oui Code anomalie pour le message d'erreur WebSphere MQ.

Table IBMQREP_DELTOMB

La table IBMQREP_DELTOMB est une table interne utilisée par le programme Q Apply pour enregistrer les suppressions conflictuelles dans la réplication entre homologues. Cette table est élaguée par le programme Q Apply.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Index créé sans clause UNIQUE : TARGET_NAME, TARGET_OWNER, VERSION_TIME DESC, KEY_HASH

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 86 fournit une brève description des colonnes dans la table IBMQREP_DELTOMB.

Tableau 86. Colonnes dans la table IBMQREP_DELTOMB

Nom de colonne	Description
TARGET_OWNER	Type de données : VARCHAR(128) ; Null admis : non Nom du propriétaire de la table cible pour laquelle la suppression conflictuelle a été enregistrée
TARGET_NAME	Type de données : VARCHAR(128) ; NULL admis : Non Nom de la table cible pour laquelle la suppression conflictuelle a été enregistrée
VERSION_TIME	Type de données : TIMESTAMP; Null admis : Non Horodatage de la suppression conflictuelle sur le serveur d'origine.
VERSION_NODE	Type de données : SMALLINT ; NULL admis : non Identifie le serveur dans un groupe d'égal à égal d'où est issue la suppression conflictuelle.
KEY_HASH	Type de données : INTEGER ; NULL admis : Non Valeur hachée de la clé pour la suppression conflictuelle.
PACKED_KEY	Type de données : VARCHAR(4096) ; NULL admis : non Valeur de la clé condensée de la suppression conflictuelle.

Table IBMQREP_DONEMSG

La table IBMQREP_DONEMSG est une table interne utilisée par le programme Q Apply pour enregistrer tous les messages de transaction ou d'administration qui ont été reçus.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Clé primaire : RECVQ, MGMSGID

Important : Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Les enregistrements dans cette table permettent de s'assurer que les messages ne sont pas traités plusieurs fois (par exemple dans le cas d'une défaillance du système) avant d'être supprimés. Le programme Q Apply supprime les entrées dans cette table au démarrage et pendant l'exécution normale.

Le tableau 87 fournit une brève description des colonnes dans la table IBMQREP_DONEMSG.

Tableau 87. Colonnes dans la table IBMQREP_DONEMSG

Nom de colonne	Description
RECVQ	Type de données : VARCHAR(97) ; NULL admis : non Nom de la file d'attente de réception où le message est arrivé.
MQMSGID	Type de données : CHAR(24) FOR BIT DATA ; NULL admis : non Identificateur de message WebSphere MQ du message.

IBMQREP_EXCEPTIONS, table

La table IBMQREP_EXCEPTIONS contient le code SQL et d'autres informations relatives aux modifications de lignes qui ne peuvent pas être appliquées en raison d'un conflit ou d'une erreur SQL.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Index créé sans clause UNIQUE : EXCEPTION_TIME

La valeur des champs SQLCODE, SQLSTATE et SQLERRMC n'est pas NULL pour les lignes qui ont été annulées par le programme Q Apply.

La taille de cette table dépend du nombre de conflits ou d'erreurs que vous prévoyez. Vous pouvez utiliser le langage SQL pour supprimer de la table les lignes inutiles.

Le tableau 88, à la page 548 fournit une brève description des colonnes dans la table IBMQREP_EXCEPTIONS.

Tableau 88. Colonnes dans la table *IBMQREP_EXCEPTIONS*

Nom de colonne	Description
EXCEPTION_TIME	Type de données : TIMESTAMP ; NULL admis : non, avec valeur par défaut Horodatage local sur le serveur Q Apply lorsqu'une erreur ou un conflit s'est produit. Valeur par défaut : horodatage actuel
RECVQ	Type de données : VARCHAR(48) ; NULL admis : non Nom de la file d'attente de réception où le message de transaction est arrivé.
SRC_COMMIT_LSN	Type de données : VARCHAR(48) FOR BIT DATA, VARCHAR(10) FOR BIT DATA pour z/OS version 7 ; NULL admis : non LSN logique sur le serveur Q Capture pour la transaction.
SRC_TRANS_TIME	Type de données : TIMESTAMP; Null admis : Non Horodatage GMT sur le serveur Q Capture lorsque la transaction a été validée dans la base de données source.
SUBNAME	Type de données : VARCHAR(128) ; NULL admis : Non Nom de l'abonnement Q auquel appartient la transaction.

Tableau 88. Colonnes dans la table IBMQREP_EXCEPTIONS (suite)

Nom de colonne	Description
REASON	<p>Type de données : CHAR(12) ; NULL admis : non</p> <p>Description de l'erreur ou du conflit qui a entraîné l'enregistrement de la transaction dans la table IBMQREP_EXCEPTIONS.</p> <p>NOTFOUND Tentative de suppression ou de mise à jour d'une ligne qui n'existe pas.</p> <p>DUPLICATE Tentative d'insertion d'une ligne déjà existante.</p> <p>CHECKFAILED La règle de détection de conflit devait vérifier toutes les valeurs ou vérifier les valeurs modifiées alors qu'une valeur non clé n'était pas prévue.</p> <p>LOBXMLTOOBIG Une valeur LOB (Large Object) ou un document XML était trop volumineux pour passer dans un message de transaction. La colonne TEXT spécifie le type de données qui était trop volumineux.</p> <p>SQLERROR Une erreur SQL qui n'était pas sur la liste des erreurs acceptables dans la colonne OKSQLSTATES de la table IBMQREP_TARGETS est survenue.</p> <p>OKSQLSTATE Une erreur SQL qui était sur la liste des erreurs acceptables dans la colonne OKSQLSTATES de la table IBMQREP_TARGETS est survenue.</p> <p>P2PDUPKEY Dans une réplication entre homologues, une mise à jour de clé a échoué car une ligne cible plus récente avec la même clé existe déjà.</p> <p>P2PNOTFOUND Dans une réplication entre homologues, une suppression ou une mise à jour a échoué car la ligne cible n'existe pas.</p> <p>P2PVERLOSER Dans une réplication entre homologues, une suppression ou une mise à jour a échoué car la ligne cible est plus récente que la ligne dans le message modifié.</p> <p>P2PINSERTED Dans une réplication entre homologues, une mise à jour de clé a été appliquée en tant qu'insertion avec succès car la ligne de l'ancienne clé et la ligne de la nouvelle clé n'ont pas été trouvées. La ligne de la nouvelle clé a été insérée dans la table cible.</p>
SQLCODE	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Code SQL renvoyé par DB2 pour la transaction.</p>
SQLSTATE	<p>Type de données : CHAR(5) ; NULL admis : oui</p> <p>Numéro d'état SQL renvoyé par DB2 pour la transaction.</p>
SQLERRMC	<p>Type de données : CHAR(70) FOR BIT DATA ; NULL admis : oui</p> <p>Jetons de messages d'erreur provenant de la structure SQLCA utilisés pour l'exécution de la transaction.</p>

Tableau 88. Colonnes dans la table IBMQREP_EXCEPTIONS (suite)

Nom de colonne	Description
OPERATION	<p>Type de données : VARCHAR(18) ; NULL admis : non</p> <p>Type d'opération SQL qui a échoué. Les valeurs possibles sont INSERT, INSERT(LOAD), DELETE, DELETE(LOAD), UPDATE, UPDATE(LOAD), KEY UPDATE, KEY UPDATE(LOAD).</p>
TEXT	<p>Type de données : CLOB ; NULL admis : non</p> <p>Instruction SQL décrivant la ligne qui a provoqué l'erreur.</p>
IS_APPLIED	<p>Type de données : CHAR(1); Null admis : Non</p> <p>Indicateur signalant si la ligne a été appliquée à la table cible même si elle a été entrée dans la table IBMQREP_EXCEPTIONS.</p> <p>Y La ligne a été appliquée car l'action de conflit définie pour l'abonnement Q était F (force).</p> <p>N La transaction n'a pas été appliquée.</p>
CONFLICT_RULE	<p>Type de données : CHAR(1) ; NULL admis : oui</p> <p>Type de détection de conflit qui a entraîné l'entrée de la ligne dans la table IBMQREP_EXCEPTIONS.</p> <p>K Seules les valeurs clé ont été vérifiées.</p> <p>C Les valeurs non clé et les valeurs clé ont été vérifiées.</p> <p>A Toutes les valeurs ont été vérifiées.</p>
SRC_TRANS_ID	<p>Type de données : VARCHAR(48) ; NULL admis : oui</p> <p>Identificateur de la transaction à laquelle appartient la ligne n'ayant pas pu être appliquée.</p>

Table IBMQREP_RECVQUEUES

La table IBMQREP_RECVQUEUES contient des informations relatives aux files d'attente locales WebSphere MQ utilisées par un programme Q Apply pour recevoir des transactions de la source. Chaque programme Q Apply peut fonctionner avec plusieurs files d'attente de réception. Chaque file d'attente de réception est identifiée de manière unique par une ligne dans la table des files d'attente de réception Q Apply.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Clé primaire : RECVQ

Index à entrées uniques : REPQMAPNAME

Important : ne modifiez pas cette table en utilisant SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 89, à la page 551 fournit une brève description des colonnes dans la table IBMQREP_RECVQUEUES.

Tableau 89. Colonnes dans la table IBMQREP_RECVQUEUES

Nom de colonne	Description
REPQMAPNAME	<p>Type de données : VARCHAR(128) ; NULL admis : non</p> <p>Nom de la mappe de files pour la réplication qui comprend cette file d'attente de réception.</p>
RECVQ	<p>Type de données : VARCHAR(48) ; NULL admis : non</p> <p>Nom de la file d'attente de réception pour cet abonnement Q.</p>
SENDQ	<p>Type de données : VARCHAR(48) ; NULL admis : oui</p> <p>Nom de la file d'attente d'envoi utilisée par le programme Q Capture pour cet abonnement Q.</p>
ADMINQ	<p>Type de données : VARCHAR(48) ; NULL admis : non</p> <p>Nom de la file d'attente d'administration utilisée par le programme Q Apply pour envoyer des messages de contrôle et d'erreur au programme Q Apply.</p>
NUM_APPLY_AGENTS	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Nombre d'unités d'exécution agent que le programme Q Apply utilise pour appliquer simultanément des transactions de cette file d'attente de réception. Une valeur égale à 1 nécessite que les transactions soient exécutées dans l'ordre dans lequel elles ont été reçues de la table source. Par défaut : 16. Nombre maximal d'unités d'exécution agent : 128.</p>
MEMORY_LIMIT	<p>Type de données : INTEGER ; NULL admis : non, avec valeur par défaut</p> <p>Quantité maximale de mémoire en méga-octet que le programme Q Apply peut utiliser comme mémoire tampon pour les messages qu'il obtient de cette file d'attente de réception. La valeur par défaut est 32 Mo; la valeur maximale est de 2 Go.</p>
CAPTURE_SERVER	<p>Type de données : VARCHAR(18) ; NULL admis : non</p> <p>Nom de la base de données où le programme Q Capture qui utilise cette file d'attente de réception s'exécute.</p> <p>z/OS Il s'agit d'un nom d'emplacement.</p>
CAPTURE_ALIAS	<p>Type de données : VARCHAR(8) ; NULL admis : non</p> <p>Alias de la base de données DB2 qui correspond au serveur Q Capture nommé dans la colonne CAPTURE_SERVER.</p>
CAPTURE_SCHEMA	<p>Type de données : VARCHAR(128) ; NULL admis : non, avec valeur par défaut</p> <p>Schéma du programme Q Capture qui utilise cette file d'attente de réception. Valeur par défaut : ASN</p>
STATE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur qui montre l'état actuel de la file d'attente de réception.</p> <p>A (par défaut) Active : le programme Q Apply traite et applique les transactions de cette file d'attente.</p> <p>I Inactive : une erreur grave a été rencontrée dans la file d'attente.</p>
STATE_TIME	<p>Type de données : TIMESTAMP ; NULL admis : non, avec valeur par défaut</p> <p>Horodatage sur le serveur Q Apply de la dernière modification d'état pour cette file d'attente de réception. Valeur par défaut : horodatage Current</p>

Tableau 89. Colonnes dans la table IBMQREP_RECVQUEUES (suite)

Nom de colonne	Description
STATE_INFO	<p>Type de données : CHAR(8) ; NULL admis : oui</p> <p>Numéro du message ASN relatif à l'état de la file d'attente. Pour plus de détails, voir la table IBMQREP_APPLYTRACE ou le journal de diagnostic Q Apply.</p>
DESCRIPTION	<p>Type de données : VARCHAR(254) ; NULL admis : oui</p> <p>Description fournie par l'utilisateur de la mappe de files de réplication contenant cette file d'attente de réception.</p>
SOURCE_TYPE	<p>Type de données : CHAR(1) ; NULL admis : oui</p> <p>La valeur de cet attribut indique le type de la source de données de chaque enregistrement.</p> <p>C Source de données Classic</p> <p>D Source de données DB2</p>
MAXAGENTS_CORRELID	<p>Type de données : INTEGER ; NULL admis : oui</p> <p>Nombre maximum d'agents Q Apply qui peuvent simultanément appliquer des transactions ayant le même <i>ID de corrélation</i>. L'ID de corrélation identifie les transactions démarrées depuis le même travail z/OS sur le serveur Q Capture. Vous pouvez utiliser la valeur MAXAGENTS_CORRELID pour limiter le parallélisme pour les travaux par lots susceptibles d'avoir de nombreuses dépendances qui pourraient provoquer un conflit d'accès. Vous pouvez définir la valeur en utilisant les commandes CREATE REPLQMAP ou ALTER REPLQMAP.</p> <p>La valeur de la colonne MAXAGENTS_CORRELID ne peut pas être supérieure à la valeur de NUM_APPLY_AGENTS. Si la valeur de MAXAGENTS_CORRELID est 1, les transactions seront appliquées en série. Si la valeur est supérieure à, par exemple 4, les quatre premières transactions seront appliquées en parallèle et les transactions suivantes seront marquées comme dépendantes de l'une des transactions. Lorsqu'une transaction se termine, une transaction dépendante ayant le même ID de corrélation est appliquée.</p>
BROWSER_THREAD_ID	<p>Type de données : VARCHAR(9) ; NULL admis : Oui</p> <p>Identificateur de corrélation pour l'unité d'exécution qui traite cette file d'attente de réception.</p>

Table IBMQREP_SAVERI

La table IBMQREP_SAVERI est une table interne que le programme Q Apply utilise pour sauvegarder les informations relatives aux contraintes d'intégrité référentielle pour les tables cible. Le programme Q Apply déplace des contraintes d'intégrité référentielle pendant le chargement des tables cible. Les contraintes sont sauvegardées dans cette table de contrôle puis restaurées après le chargement des tables.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 90 fournit une brève description des colonnes dans la table IBMQREP_SAVERI.

Tableau 90. Colonnes dans la table IBMQREP_SAVERI

Nom de colonne	Description
SUBNAME	Type de données : VARCHAR(132) ; NULL admis : non Nom de l'abonnement Q auquel appartiennent les tables cible.
RECVQ	Type de données : VARCHAR(48) ; NULL admis : non Nom de la file d'attente de réception définie pour l'abonnement Q.
CONSTNAME	Type de données : VARCHAR (128), VARCHAR(18) pour z/OS version 7 ; NULL admis : non Nom unique de la contrainte.
TABSCHEMA	Type de données : VARCHAR(128); NULL admis : Non Schéma ou le qualificatif de haut niveau de la table enfant sur laquelle la contrainte est définie.
TABNAME	Type de données : VARCHAR(128) ; NULL admis : Non Nom de la table enfant sur laquelle la contrainte est définie.
REFTABSCHEMA	Type de données : VARCHAR(128) ; NULL admis : Non Schéma de la table parente sur laquelle la contrainte est définie.
REFTABNAME	Type de données : VARCHAR(128) ; NULL admis : Non Nom de la table parente sur laquelle la contrainte est définie.
ALTER_RI_DDL	Type de données : VARCHAR(1680) ; NULL admis : non Déclaration ALTER TABLE utilisée pour restaurer les contraintes d'intégrité référentielle.
TYPE_OF_LOAD	Type de données : CHAR(1); Null admis : Non Indicateur signalant le type de phase de chargement. I Un chargement automatique. E Un chargement manuel.
DELETERULE	Type de données : CHAR(1) ; NULL admis : oui Règle de suppression définie pour la contrainte. A NO ACTION C CASCADE N SET NULL R RESTRICT
UPDATERULE	Type de données : CHAR(1) ; NULL admis : oui Règle de suppression définie pour la contrainte. A NO ACTION R RESTRICT

Table IBMQREP_SPILLQS

La table IBMQREP_SPILLQS est une table interne utilisée par le programme Q Apply pour enregistrer les files d'attente auxiliaires temporaires qu'il a créées pour conserver les messages pendant le chargement des tables cible. Le programme Q Apply supprime les files d'attente auxiliaires quand elles ne sont plus nécessaires.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Clé primaire : SPILLQ

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 91 fournit une brève description des colonnes dans la table IBMQREP_SPILLQS.

Tableau 91. Colonnes dans la table IBMQREP_SPILLQS

Nom de colonne	Description
SPILLQ	Type de données : VARCHAR(48) ; NULL admis : non Nom de la file d'attente auxiliaire temporaire utilisée pour cet abonnement Q.
SUBNAME	Type de données : VARCHAR(132) ; NULL admis : non Nom de l'abonnement Q.
RECVQ	Type de données : VARCHAR(48) ; NULL admis : non Nom de la file d'attente de réception utilisée pour cet abonnement Q.

Table IBMQREP_SPILLEDROW

La table IBMQREP_SPILLEDROW est une table interne utilisée par le programme Q Apply pour enregistrer les messages qui sont envoyés vers une file d'attente auxiliaire temporaire pendant le chargement des cibles.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Clé primaire : SPILLQ, MQMSGID

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le programme Q Apply supprime des lignes dans cette table une fois que les messages qu'elles représentent ont été pris de la file d'attente auxiliaire et appliqués à la table cible.

Le tableau 92, à la page 555 fournit une brève description des colonnes dans la table IBMQREP_SPILLEDROW.

Tableau 92. Colonnes dans la table *IBMQREP_SPILLEDROW*

Nom de colonne	Description
SPILLQ	Type de données : VARCHAR2(48) ; NULL admis : non Nom de la file d'attente auxiliaire où le message a été temporairement stocké.
MQMSGID	Type de données : CHAR(24) FOR BIT DATA ; NULL admis : non Identificateur de message WebSphere MQ du message.

Table **IBMQREP_TARGETS**

La table *IBMQREP_TARGETS* stocke les informations des abonnements Q pour le programme Q Apply, y compris le type et l'état, les actions d'erreur par défaut et les règles de gestion des conflits de ligne.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Index à entrées uniques : SUBNAME, RECVQ

Index créé sans clause UNIQUE : TARGET_OWNER ASC, TARGET_NAME ASC, RECVQ ASC, SOURCE_OWNER ASC, SOURCE_NAME ASC

Index créé sans clause UNIQUE : RECVQ, SUB_ID

Index créé sans clause UNIQUE : SPILLQ, STATE

Important : ne modifiez pas cette table en utilisant SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 93 fournit une brève description des colonnes dans la table *IBMQREP_TARGETS*.

Tableau 93. Colonnes dans la table *IBMQREP_TARGETS*

Nom de colonne	Description
SUBNAME	Type de données : VARCHAR(132) ; NULL admis : non Nom de l'abonnement Q. Il doit être unique pour chaque paire source-cible et ne doit pas contenir de blancs.
RECVQ	Type de données : VARCHAR(48) ; NULL admis : non Nom de la file d'attente de réception utilisée pour cet abonnement Q.
SUB_ID	Type de données : INTEGER ; NULL admis : oui Un entier généré par le programme Q Capture et utilisé pour identifier de façon unique un abonnement Q dans le message subscription schema vers le programme Q Apply.
SOURCE_SERVER	Type de données : VARCHAR(18) ; NULL admis : non Nom de la base de données ou du sous-système contenant la table source pour cet abonnement Q. Pour z/OS, il s'agit d'un nom d'emplacement. Valeur de l'entrée <i>source de données</i> du fichier de configuration ASNCLP servant à la réplication Classic, qui est le nom du processeur de requêtes sur le serveur de données Classic.

Tableau 93. Colonnes dans la table IBMQREP_TARGETS (suite)

Nom de colonne	Description
SOURCE_ALIAS	Type de données : VARCHAR(8) ; NULL admis : non L'alias de la base de données DB2 qui correspond au serveur Q Capture nommé dans la colonne SOURCE_SERVER.
SOURCE_OWNER	Type de données : VARCHAR(128) ; NULL admis : Non Nom de schéma ou qualificatif de haut niveau de la table source pour cet abonnement Q.
SOURCE_NAME	Type de données : VARCHAR(128) ; NULL admis : Non Nom de la table source pour cet abonnement Q.
SRC_NICKNAME_OWNER	Type de données : VARCHAR(128) ; NULL admis : Oui Le schéma du pseudonyme attribué à la table source pour les chargements automatiques utilisant l'utilitaire LOAD from CURSOR lorsque le programme Q Apply est exécuté sur une plateforme autre que z/OS.
SRC_NICKNAME	Type de données : VARCHAR(128) ; NULL admis : Oui Le pseudonyme attribué à la table source pour les chargements automatiques utilisant l'utilitaire LOAD from CURSOR lorsque le programme Q Apply est exécuté sur une plateforme autre que z/OS.
TARGET_OWNER	Type de données : VARCHAR(128) ; NULL admis : Non Nom de schéma ou qualificatif de haut niveau de la table source ou de la procédure stockée pour cet abonnement Q.
TARGET_NAME	Type de données : VARCHAR(128) ; NULL admis : Non Nom de la table cible pour cet abonnement Q.
TARGET_TYPE	Type de données : INTEGER ; NULL admis : non, avec valeur par défaut Indicateur signalant le type de cible de réplication. 1 (par défaut) Table utilisateur 2 Table de modification cohérente des données (CCD) 3 Réservé pour un usage ultérieur. 4 Réservé pour un usage ultérieur. 5 Procédure stockée
FEDERATED_TGT_SRVR	Type de données : VARCHAR(18) ; NULL admis : oui Nom de la base de données relationnelle contenant la cible de l'abonnement Q.

Tableau 93. Colonnes dans la table IBMQREP_TARGETS (suite)

Nom de colonne	Description
STATE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur inséré par le programme Q Apply pour décrire l'état actuel de l'abonnement Q.</p> <p>I (par défaut) Le programme Q Apply n'applique pas de modification à la cible car l'abonnement Q est nouveau ou en erreur. Le programme Q Apply élimine toutes les transactions qu'il a reçues pour l'abonnement Q et attend un nouveau message subscription schema.</p> <p>L Le programme Q Capture a commencé l'activation de l'abonnement Q en envoyant un message subscription schema et envoie des modifications provenant de la table source.</p> <p>E La table cible est en cours de chargement par une application externe. Le programme Q Apply met des messages de modification dans une file d'attente auxiliaire en attendant le chargement de la table.</p> <p>D La table cible est chargée et le programme Q Apply est prêt à envoyer un message load done au programme Q Capture. Cet état est employé uniquement pour les chargements automatiques.</p> <p>F Le programme Q Apply applique des messages provenant de la file d'attente auxiliaire.</p> <p>T Le programme Q Apply s'arrête en raison d'une erreur. Il désactive l'abonnement Q puis vide et supprime la file d'attente auxiliaire.</p> <p>A Le programme Q Apply applique des modifications à la cible.</p> <p>R Le programme Q Apply reprend des opérations après qu'un abonnement Q ait été placé dans l'état déversé (S) par le paramètre sp11sub de la commande MODIFY ou asnaqcmd, ou durant une phase de charge initiale d'abonnement Q lors de la réception d'un message load done received du programme Q Capture. Le paramètre resumesub met l'abonnement Q à l'état de reprise (R) une fois la maintenance de la table cible effectuée.</p> <p>Cet état signifie que le programme Q Apply traite des lignes qui sont dans la file d'attente auxiliaire. Tant que le programme Q Apply ne vide pas la file d'attente auxiliaire, les lignes entrantes continuent d'être déversées. Une fois la file d'attente auxiliaire vidée par le programme Q Apply, l'abonnement Q est mis à l'état actif (A) et les opérations normales peuvent reprendre.</p> <p>S Le programme Q Apply place les lignes de l'abonnement Q dans une file d'attente auxiliaire temporaire. Définir le paramètre sp11sub met l'abonnement Q à l'état déversé pour que vous puissiez effectuer des opérations de maintenance sur la table cible.</p> <p>W Pour les configurations entre homologues avec plus de deux serveurs et une phase de chargement : le programme Q Apply a vu un message de schéma d'abonnement, déverse activement les modifications sur la table source et attend une confirmation ((autre message de schéma) du programme Q Capture pour commencer le chargement.</p> <p>P Une erreur SQL a été détectée tandis que Q Apply appliquait des messages d'une file d'attente auxiliaire et que l'abonnement Q était à l'état F. L'unité d'exécution de l'agent Q Apply qui traitait la file d'attente auxiliaire s'arrête. Les modifications futures de l'abonnement Q sont placées dans la même file d'attente auxiliaire jusqu'à l'émission du paramètre resumesub des commandes MODIFY ou asnaqcmd.</p> <p>U Un état interne qui indique que le programme Q Apply a créé cet abonnement Q et que l'abonnement Q n'a pas encore été activé.</p>
STATE_TIME	<p>Type de données : TIMESTAMP ; NULL admis : non, avec valeur par défaut</p> <p>Horodatage sur le serveur Q Apply de la dernière modification d'état pour cet abonnement Q. Valeur par défaut : horodatage Current</p>

Tableau 93. Colonnes dans la table IBMQREP_TARGETS (suite)

Nom de colonne	Description
STATE_INFO	<p>Type de données : CHAR(8) ; NULL admis : oui</p> <p>Le numéro du message ASN relatif à l'état de l'abonnement Q. Pour plus de détails, voir la table IBMQREP_APPLYTRACE ou le journal de diagnostic Q Apply.</p>
SUBTYPE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant le type de réplication dans lequel l'abonnement Q est impliqué.</p> <p>U (par défaut) Réplication unidirectionnelle.</p> <p>B Réplication bidirectionnelle.</p> <p>P Réplication entre homologues.</p>
CONFLICT_RULE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant au programme Q Apply la méthode de recherche de modifications conflictuelles sur la table cible. Les insertions sont toujours vérifiées à l'aide de la règle K (contrôler uniquement les clés) car il n'y a pas de valeurs avant (before) et les clés doivent être utilisées pour détecter les conflits.</p> <p>K (par défaut) Contrôler uniquement les clés. Le programme Q Apply recherche des conflits en comparant la valeur actuelle de la clé primaire dans la table cible avec la valeur de l'ancienne clé envoyée depuis la table source.</p> <p>C Contrôler les colonnes modifiées. Avant de mettre à jour les colonnes cible, le programme Q Apply s'assure que leurs valeurs actuelles correspondent aux valeurs avant (before) dans les colonnes source. Pour les suppressions, le programme Q Apply vérifie toutes les colonnes.</p> <p>A Contrôler toutes les colonnes. Avant de mettre à jour ou de supprimer une ligne, le programme Q Apply s'assure que les valeurs actuelles dans toutes les colonnes correspondent aux anciennes valeurs dans la table source.</p> <p>V Contrôler la version. En réplication entre homologues, le programme Q Apply vérifie la colonne de version avant d'appliquer une ligne.</p>
CONFLICT_ACTION	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur qui signale au programme Q Apply ce qu'il doit faire quand il détecte un conflit sur une ligne :</p> <p>I (par défaut) Le programme Q Apply n'applique pas la ligne conflictuelle mais applique les autres lignes dans la transaction.</p> <p>F Le programme Q Apply tente de forcer la modification. Cela nécessite que le programme Q Capture envoie toutes les colonnes ; la valeur CHANGED_COLS_ONLY doit donc être égale à N (non) dans la table IBMQREP_SUBS. Il s'agit de la valeur par défaut lorsqu'une table cible est en cours de chargement.</p> <p>D Le programme Q Apply n'applique pas la ligne conflictuelle mais applique les autres lignes dans la transaction. Puis il désactive l'abonnement Q, arrête l'application des transactions à la cible et envoie un rapport d'erreur au programme Q Capture sur la file d'attente d'administration.</p> <p>S Le programme Q Apply annule la transaction, valide puis s'arrête.</p> <p>Q Le programme Q Apply arrête la lecture depuis la file d'attente.</p> <p>Toutes les lignes conflictuelles sont insérées dans la table IBMQREP_EXCEPTIONS.</p>

Tableau 93. Colonnes dans la table IBMQREP_TARGETS (suite)

Nom de colonne	Description
ERROR_ACTION	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant au programme Q Apply ce qu'il doit faire au cas où une erreur telle qu'une erreur SQL (autre qu'un conflit) l'empêche d'appliquer une modification de ligne. Cet indicateur n'a aucune influence sur le comportement du programme Q Apply pour les erreurs qui ne sont pas liées à l'application d'une modification de ligne, par exemple les erreurs WebSphere MQ liées à la lecture depuis une file d'attente.</p> <p>Q (par défaut) Le programme Q Apply arrête la lecture depuis la file d'attente.</p> <p>D Le programme Q Apply n'applique pas la ligne conflictuelle mais applique les autres lignes dans la transaction. Puis il désactive l'abonnement Q, arrête l'application des transactions à la cible et envoie un rapport d'erreur au programme Q Capture sur la file d'attente d'administration.</p> <p>S Le programme Q Apply annule la transaction, valide puis s'arrête.</p> <p>B Le programme Q Apply place les messages de modification de l'abonnement Q dans une file d'attente auxiliaire temporaire au cours de la correction de l'erreur SQL. Utilisez le paramètre resumesub de la commande MODIFY ou asnqacmd pour inviter Q Apply à démarrer l'application de messages issus de la file d'attente auxiliaire aux cibles. Pour effectuer cette opération, vous devez spécifier une file d'attente de modèles pour l'abonnement Q lorsque vous le créez ou le modifier à l'aide d'outils d'administration de réplication. Cette action sur l'erreur n'est pas prise en charge pour les tables ayant des contraintes d'intégrité référentielle.</p> <p>Toutes les lignes conflictuelles sont insérées dans la table IBMQREP_EXCEPTIONS.</p>
SPILLQ	<p>Type de données : VARCHAR(48) ; NULL admis : oui</p> <p>Nom de la file d'attente auxiliaire temporaire créée par le programme Q Apply au moment du chargement des cibles.</p>
OKSQLSTATES	<p>Type de données : VARCHAR(128) ; NULL admis : Oui</p> <p>Une liste de valeurs SQLSTATE séparées par un espace que le programme Q Apply ne considère pas comme des erreurs. Vous spécifiez ces valeurs lorsque vous définissez un abonnement Q.</p> <p>Les valeurs saisies pour OKSQLSTATES indiquent au programme Q Apply d'ignorer l'action d'erreur définie pour l'abonnement Q. Les valeurs OKSQLSTATES n'affectent pas les conflits tels que les doublons et les erreurs de lignes introuvables, qui sont gérés par l'action de conflit définie pour l'abonnement Q.</p> <p>Restriction : La fonction OKSQLSTATES n'est pas prise en charge lorsque toutes les conditions suivantes sont vraies :</p> <ul style="list-style-type: none"> • L'abonnement Q utilise des expressions. • Au moins une colonne de clé est utilisée dans l'expression. • L'évaluation de l'expression échoue à cause de données erronées ou pour une autre raison.
SUBGROUP	<p>Type de données : VARCHAR(30); Null admis : Oui</p> <p>Nom du groupe de réplication entre homologues comprenant cet abonnement Q.</p>
SOURCE_NODE	<p>Type de données : SMALLINT ; NULL admis : non, avec valeur par défaut</p> <p>Un numéro d'identification pour le serveur source dans un abonnement Q entre homologues. Par défaut : 0</p>
TARGET_NODE	<p>Type de données : SMALLINT ; NULL admis : non, avec valeur par défaut</p> <p>Un numéro d'identification pour le serveur cible dans un abonnement Q entre homologues. Par défaut : 0</p>

Tableau 93. Colonnes dans la table *IBMQREP_TARGETS* (suite)

Nom de colonne	Description
GROUP_INIT_ROLE	<p>Type de données : CHAR(1) ; NULL admis : oui</p> <p>Le rôle de ce serveur cible dans le processus d'initialisation d'un abonnement Q entre homologues.</p> <p>I L'initiateur du groupe entre homologues où le signal CAPSTART est entré dans la table IBMQREP_SIGNAL pour initialiser l'abonnement.</p> <p>M Un serveur dans le groupe entre homologues qui n'est pas utilisé pour initialiser l'abonnement.</p> <p>N Un nouveau serveur qui est dans le processus de jonction d'un groupe entre homologues.</p>
HAS_LOADPHASE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si la table cible sera chargée avec des données de la source.</p> <p>N (par défaut) La cible ne sera pas chargée.</p> <p>I Un chargement automatique. Le programme Q Apply charge la table cible.</p> <p>E Un chargement manuel. Une autre application que le programme Q Apply charge la table cible.</p>

Tableau 93. Colonnes dans la table IBMQREP_TARGETS (suite)

Nom de colonne	Description
LOAD_TYPE	<p>Type de données : SMALLINT ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant quel utilitaire est appelé pour le chargement de la table cible lorsque HAS_LOADPHASE est sur I (chargement automatique).</p> <p>0 (par défaut) Le programme Q Apply sélectionne l'utilitaire de chargement parmi les choix ci-dessous.</p> <p>1 Utiliser l'utilitaire LOAD from CURSOR. L'utilitaire est appelé avec une option afin de supprimer toutes les données de la table cible avant de les remplacer par les données de la source (c'est ce qu'on appelle l'option de remplacement).</p> <p>101 Utiliser l'utilitaire LOAD from CURSOR. L'utilitaire est appelé avec une option pour ajouter des données source à la table cible sans supprimer le contenu de la table cible. C'est ce qu'on appelle l'option reprise sur les cibles z/OS et l'option insérer sur les cibles Linux, UNIX et Windows.</p> <p>2 Utiliser les utilitaires EXPORT et IMPORT. Les utilitaires sont appelés avec une option afin de supprimer toutes les données de la table cible avant de les remplacer par les données de la source (c'est ce qu'on appelle l'option de remplacement).</p> <p>102 Cibles Linux, UNIX et Windows : Utilisez les utilitaires IMPORT et EXPORT. L'utilitaire LOAD est appelé avec une option pour ajouter des données source à la table cible sans supprimer le contenu de la table cible (c'est ce qu'on appelle l'option insérer).</p> <p>3 Utiliser les utilitaires EXPORT et LOAD. Les utilitaires sont appelés avec une option afin de supprimer toutes les données de la table cible avant de les remplacer par les données de la source (c'est ce qu'on appelle l'option de remplacement).</p> <p>103 Cibles Linux, UNIX et Windows : Utilisez les utilitaires IMPORT et LOAD. L'utilitaire LOAD est appelé avec une option pour ajouter des données source à la table cible sans supprimer le contenu de la table cible (c'est ce qu'on appelle l'option insérer).</p> <p>4 Sélectionnez une source de réplication et utilisez l'utilitaire LOAD ou, pour les cibles Oracle, exécutez l'utilitaire SQL*Loader (réplication unidirectionnelle uniquement). Les utilitaires sont appelés avec une option afin de supprimer toutes les données de la table cible avant de les remplacer par les données de la source (option de remplacement). Cibles Oracle : Pour utiliser SQL*Loader, vous devez créer un fichier de mots de passe à l'aide de la commande asnpwd dans le répertoire spécifié par le paramètre apply_path ou le répertoire depuis lequel Q Apply est appelé avec les valeurs suivantes pour ces mots-clé :</p> <ul style="list-style-type: none"> • alias : L'entrée Oracle tnsnames.ora fait référence au serveur Oracle (nom identique utilisé avec l'option NODE de la commande CREATE SERVER pour configurer la fédération). • id : ID utilisateur distant permettant de se connecter à la base de données Oracle. • password : mot de passe permettant de se connecter à la base de données Oracle. <p>Le fichier doit s'appeler asnpwd.aut par défaut. Avant de commencer l'abonnement Q, testez la connectivité avec cette commande : <code>\$> sqlplus id/mot de passe@alias</code>.</p> <p>104 Sélectionnez une source de réplication et utilisez l'utilitaire LOAD ou, pour les cibles Oracle, l'utilitaire SQL*Plus. Les utilitaires sont appelés avec une option pour ajouter des données source à la table cible sans supprimer le contenu de la table cible (option reprendre ou insérer). Pour utiliser SQL*Plus, suivez les instructions dans l'entrée de LOAD_TYPE 4 ci-dessus.</p>

Tableau 93. Colonnes dans la table IBMQREP_TARGETS (suite)

Nom de colonne	Description
LOAD_TYPE (suite)	<p>5 Cibles Linux, UNIX et Windows : Sélectionnez une source de réplication et utilisez l'utilitaire IMPORT. L'utilitaire est appelé avec une option afin de supprimer toutes les données de la table cible avant de les remplacer par les données de la source (c'est ce qu'on appelle l'option de remplacement).</p> <p>105 Cibles Linux, UNIX et Windows : Sélectionnez une source de réplication et utilisez l'utilitaire IMPORT. L'utilitaire est appelé avec une option pour ajouter des données source à la table cible sans supprimer le contenu de la table cible (c'est ce qu'on appelle l'option insérer).</p>
DESCRIPTION	<p>Type de données : VARCHAR(254) ; NULL admis : oui</p> <p>Une description fournie par l'utilisateur de l'abonnement Q.</p>
SEARCH_CONDITION	<p>Type de données : VARCHAR(2048) ; NULL admis : oui</p> <p>La condition de recherche utilisée pour filtrer les lignes pour l'abonnement Q. Il doit s'agir d'une clause WHERE de l'instruction select annotée avec deux points directement devant les noms des colonnes source.</p>
MODELQ	<p>Type de données : VARCHAR(36) ; NULL admis : oui, avec valeur par défaut</p> <p>Nom de la file d'attente modèle que le programme Q Apply utilise pour créer les files d'attente auxiliaires pendant le processus de chargement de la cible. Par défaut : IBMQREP.SPILL.MODELQ</p>
CCD_CONDENSED	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si une table cible CCD est condensée ou non.</p> <p>N (par défaut) La table CCD est non condensée, ce qui signifie qu'elle contient plusieurs lignes ayant la même valeur de clé, une ligne pour chaque changement se produisant sur la table source.</p> <p>Y La table CCD est condensée, ce qui signifie qu'elle contient une ligne par valeur de clé dans la table source et contient uniquement la dernière valeur pour la ligne.</p>
CCD_COMPLETE	<p>Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut</p> <p>Indicateur signalant si une table cible CCD est complète ou incomplète.</p> <p>N (par défaut) La table CCD est incomplète, ce qui signifie qu'elle contient uniquement des modifications sur la table source et démarre sans données.</p> <p>Y La table CCD est complète, ce qui signifie qu'elle contient toutes les lignes présentant un intérêt provenant de la table source et est initialisée avec un ensemble complet de données source.</p>
SOURCE_TYPE	<p>Type de données : CHAR(1) ; NULL admis : oui</p> <p>La valeur de cet attribut indique le type de la source de données de chaque enregistrement.</p> <p>C Source de données Classic</p> <p>D Source de données DB2</p>
SCHEMA_SUBNAME	<p>Type de données : VARCHAR(64) ; NULL admis : oui</p> <p>Le nom de l'abonnement Q de niveau schéma si cet abonnement Q de niveau table a été créé par le programme Q Apply ou le programme de ligne de commande ASNCLP. Sinon, la valeur est NULL.</p>
SUB_CREATOR	<p>Type de données : VARCHAR(12) ; NULL admis : oui</p> <p>Identifie la manière dont cet abonnement Q a été créé. Les valeurs sont "asnclp", "replcenter" (le Centre de réplication) et "asnqapp" (le programme Q Apply). Lorsque les tables de contrôle de Q Apply sont migrées d'une version à la version 10 sous Linux, UNIX et Windows, la valeur de la colonne est NULL.</p>

Table IBMQREP_TRG_COLS

La table IBMQREP_TRG_COLS identifie le mappage entre une colonne dans la table source et une colonne dans la table cible ou entre la colonne source et un paramètre si la cible est une procédure stockée.

Serveur : serveur Q Apply

Schéma par défaut : ASN

Index à entrées uniques : RECVQ, SUBNAME, TARGET_COLNAME

Important : ne modifiez pas cette table à l'aide de SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le programme Q Apply entre des valeurs dans cette table selon les informations reçues dans le message de schéma du programme Q Capture.

Le tableau 94 fournit une brève description des colonnes dans la table IBMQREP_TRG_COLS.

Tableau 94. Colonnes dans la table IBMQREP_TRG_COLS

Nom de colonne	Description
RECVQ	Type de données : VARCHAR(48) ; NULL admis : non Nom de la file d'attente de réception utilisée pour cet abonnement Q.
SUBNAME	Type de données : VARCHAR(132) ; NULL admis : non Nom de l'abonnement Q.
SOURCE_COLNAME	Type de données : VARCHAR(1024) ; NULL admis : Non Cette colonne peut contenir l'une des valeurs suivantes : <ul style="list-style-type: none">• Nom de la colonne source à répliquer ou publier• Expression SQL ou XML utilisée pour créer le contenu de la colonne cible• Colonne d'audit dans une table de modification cohérente des données
TARGET_COLNAME	Type de données : VARCHAR(128) ; NULL admis : Non Nom de la colonne cible. Si la cible est une procédure stockée, cette colonne contient le nom du paramètre auquel le programme Q Apply transmet la valeur de la colonne source.
TARGET_COLNO	Type de données : INTEGER ; NULL admis : oui Nombre attribué à une colonne cible. Si la cible est une procédure stockée, cette colonne contient un nombre attribué au paramètre auquel le programme Q Apply transmet la valeur de la colonne source.
MSG_COL_CODEPAGE	Type de données : INTEGER ; NULL admis : oui Identificateur pour la page de codes utilisée pour coder la valeur de la colonne source.
MSG_COL_NUMBER	Type de données : SMALLINT ; NULL admis : Oui Ordre de présentation de la colonne source dans un message de modification, commençant par 0.

Tableau 94. Colonnes dans la table IBMQREP_TRG_COLS (suite)

Nom de colonne	Description
MSG_COL_TYPE	Type de données : SMALLINT ; NULL admis : Oui Type de données DB2 de la colonne cible.
MSG_COL_LENGTH	Type de données : SMALLINT ; NULL admis : Oui Longueur maximale de donnée définie sur la colonne cible.
IS_KEY	Type de données : CHAR(1); Null admis : Non Indicateur signalant si la colonne source fait partie de la clé pour la table source. Si la valeur de cet indicateur ne correspond pas à la définition des clés de table, le programme Q Apply rejette le message de schéma et invalide l'abonnement Q : Y La colonne fait partie de la clé de table source. N La colonne ne fait pas partie de la clé de table source. Restriction : Les colonnes d'objets LOB (Large object) et les colonnes LONG ne peuvent pas être utilisées dans la clé de publication ou de réplication.
MAPPING_TYPE	Type de données : CHAR(1) ; NULL admis : non, avec valeur par défaut Indicateur signalant le type de mappage entre la colonne source et la colonne cible. R (par défaut) Un mappage normal où la valeur dans la colonne source correspond à la valeur dans la colonne cible. E Expression SQL utilisée pour créer le contenu de la colonne cible. A Colonne d'audit dans une table de modification cohérente des données. La colonne n'existe pas dans la table source.
SRC_COL_MAP	Type de données : VARCHAR(2000) ; NULL admis : oui, avec valeur par défaut Position, type de données, longueur et page de codes de toutes les colonnes utilisées dans une expression SQL. Valeur par défaut : NULL
BEF_TARG_COLNAME	Type de données : VARCHAR(128) ; NULL admis : oui, avec valeur par défaut Définit le nom de colonne de l'image avant s'il en existe une (pour les cibles CCD uniquement). Valeur par défaut : NULL

Structure détaillée des colonnes de gestion des versions pour la réplication entre homologues

Les tables participant à la réplication entre homologues ont besoin de deux colonnes de gestion des versions, une colonne d'horodatage ou une colonne de chiffre entier, gérées par des déclencheurs. Ces deux colonnes permettent aux programmes Q Capture et Q Apply d'effectuer le contrôle de conflit basé sur la version nécessaire à la réplication entre homologues.

Les colonnes permettent également aux deux programmes de résoudre des conflits afin de maintenir la convergence des tables dans un groupe d'abonnement Q. Les valeurs dans ces colonnes reflètent la version la plus courante d'une ligne.

Ces colonnes et ces déclencheurs supplémentaires sont créés lorsque vous utilisez le programme de ligne de commande ASNCLP ou le Centre de réplication pour

créer des abonnements Q pour la réplication entre homologues. Lorsque vous créez un abonnement Q pour la réplication entre homologues, vous devez vous abonner à ces deux colonnes.

Le tableau 95 fournit une brève description des colonnes supplémentaires dans les tables utilisateur nécessaires à la réplication entre homologues.

Tableau 95. Colonnes supplémentaires nécessaires à la réplication entre homologues

Nom de colonne	Description
IBMQREPVERNODE	Type de données : SMALLINT ; NULL admis : non, avec valeur par défaut Un nombre identifiant la base de données ou le sous-système qui contient la table dans un groupe d'égal à égal. Par défaut : 0
IBMQREPVERTIME	Type de données : TIMESTAMP ; NULL admis : non, avec valeur par défaut Un horodatage enregistrant une modification sur la table. Par défaut : 0001-01-01-00.00.00

Chapitre 26. Structure des messages XML pour la publication d'événements

Dans la publication d'événements, un programme Q Capture et une application utilisateur échangent des messages XML (Extensible Markup Language).

Les rubriques suivantes décrivent en détail les messages XML qui sont échangés, ainsi que leur structure :

Types de message XML et conditions requises

Un programme Q Capture utilise des messages XML pour envoyer des transactions ou des modifications au niveau des lignes à une application utilisateur. Le programme Q Capture et l'application utilisateur se servent aussi de messages XML pour communiquer.

Les rubriques suivantes décrivent les types de messages XML et les conditions techniques requises :

Types de message

Un programme Q Capture envoie des messages de données et des messages d'information à une application utilisateur, qui envoie des messages de contrôle au programme Q Capture.

Le tableau 96 décrit les trois types de messages.

Tableau 96. Messages envoyés par un programme Q Capture et une application utilisateur

Type de message	Sens	Description
Données	Q Capture vers l'application utilisateur	Contient l'un des éléments suivants de la table source : <ul style="list-style-type: none">• tout ou partie d'une transaction,• une opération de ligne unique,• tout ou partie d'une valeur d'objet LOB à partir d'une opération de ligne dans une transaction.
Information	Q Capture vers l'application utilisateur	Fournit des informations sur l'état du programme Q Capture ou d'une publication.
Contrôle	Application utilisateur vers Q Capture	Demande au programme Q Capture d'activer ou de désactiver une publication, d'invalider une file d'attente d'envoi ou de confirmer le chargement d'une table cible.

Conditions techniques requises pour les messages XML

Le programme Q Capture génère des messages sous la forme d'instances de documents XML conformément aux recommandations suivantes.

- Les messages sont codés en Unicode à l'aide du code UTF-8 (page de codes 1208) comme indiqué dans la recommandation W3C sur XML 1.0 (2e édition) du 6 octobre 2000.

- La structure des messages suit la recommandation W3C sur le langage du schéma XML (Part 1: Structure and Part 2: Datatypes) du 2 mai 2001.

Les modifications provenant de la base de données source sont converties en messages à l'aide de la version d'ICU4C (International Components for Unicode) d'IBM qui est livrée avec DB2.

Pour interpréter les messages de contrôle provenant de l'application d'abonnement, le programme Q Capture utilise l'analyseur syntaxique XML d'IBM, XML4C version 5.3.

Traitement des délimiteurs XML dans les données de type caractères

Dans les messages de données d'un programme Q Capture, les valeurs des colonnes abonnées figurent entre des balises XML qui décrivent le type de données des colonnes.

Par exemple, les valeurs 222 et Hello d'une table source seraient codées sous la forme `<integer>222</integer>` et `<varchar>Hello</varchar>`.

Comme les signes supérieur ou inférieur (< ou >) et le perluète (&) sont des délimiteurs XML prédéfinis, le programme Q Capture les convertit comme suit lorsqu'ils apparaissent dans des valeurs de colonne :

- < en `<`;
- > en `>`;
- & en `&`;

De même, lorsque l'apostrophe (') ou les guillemets (") apparaissent dans les valeurs d'attribut, le programme Q Capture les convertit comme suit :

- ' en `'`;
- " en `"`;

Les messages résultants sont des instances de document XML valides.

Structure des messages provenant de Q Capture vers une application utilisateur

Un programme Q Capture envoie des messages de données et des messages d'information à une application utilisateur. Les messages de données apportent des modifications à une table source qui fait partie d'une publication. Les messages d'information confirment la demande d'une application utilisateur via un message de contrôle, signalent l'état du programme Q Capture ou des erreurs de publication.

Liste des messages provenant de Q Capture vers une application utilisateur

Le programme Q Capture envoie deux types de messages à une application utilisateur : des messages de données et des messages de contrôle.

Messages de données

Contiennent les modifications apportées à une table source. Le tableau 97, à la page 569 contient une référence rapide des différents types de messages de données.

Messages d'information

Signalent l'état d'un programme Q Capture ou d'une publication. Le tableau 98 contient une référence rapide aux types de messages d'information.

Tableau 97. Messages de données envoyés par le programme Q Capture à une application utilisateur

Type de message	Description
Transaction	Contient une ou plusieurs opérations d'insertion, de suppression ou de mise à jour d'une table source. Ces opérations dépendent de la même transaction de base de données. Contient également des informations de validation pour la transaction.
Opération de ligne	Contient une opération unique d'insertion, de suppression ou de mise à jour d'une table source. Contient également des informations de validation sur la transaction de base de données dont dépend cette ligne.
Objet LOB	Contient tout ou partie des données d'une valeur LOB dans la table source. Les messages LOB sont envoyés séparément des messages de transaction et des messages d'opération de ligne dont dépendent les valeurs LOB.

Tableau 98. Messages d'information envoyés par le programme Q Capture à une application utilisateur

Type de message	Description
Abonnement désactivé	Indique à l'application utilisateur que le programme Q Capture a désactivé une publication.
Chargement terminé reçu	Confirme que le programme Q Capture a reçu le message indiquant que la table cible est chargée.
Rapport d'erreurs	Indique à l'application utilisateur que le programme Q Capture a rencontré une erreur de publication.
Pulsation	Indique à l'application utilisateur que le programme Q Capture est toujours en cours d'exécution même s'il n'a aucun message de données à envoyer.
Schéma d'abonnement	Contient des informations sur la table source et ses colonnes. Contient également des options d'envoi de données, le nom de la file d'attente d'envoi et des informations sur le programme Q Capture et la base de données source.
Ajout de colonne	Contient des informations sur une colonne ajoutée à une publication existante.

msg : Élément principal des messages XML provenant du programme Q Capture vers une application utilisateur

L'élément msg est l'élément racine de tous les messages de données et d'information envoyés par le programme Q Capture à une application utilisateur.

Le tableau 99, à la page 570 décrit l'élément msg.

Tableau 99. Description d'élément pour l'élément msg (du programme Q Capture à l'application utilisateur)

Nom	Propriétés
msg	Contenu complexe de type complexe non vide

Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="instance_schéma_XML"
      xsi:noNamespaceSchemaLocation="document_schéma"
      version="version" dbName="nom_base-données">

    elements

</msg>
```

Détails

instance_schéma_XML

URL de l'instance de schéma XML. Pour la publication d'événements, cette URL est www.w3.org/2001/XMLSchema-instance. Type de données XML : chaîne.

document_schéma

Nom de fichier du document de schéma XML. L'espace de nom XML n'est pas pris en charge dans la publication d'événements parce que les messages ne se rapportent qu'à un seul schéma XML. Les messages envoyés par un programme Q Capture à une application utilisateur se rapportent au document de schéma mqcap.xsd. Type de données XML : chaîne.

version

Version du schéma de message XML. Type de données XML : chaîne.

nom_base-données

Nom de la base de données ou du sous-système source. Type de données XML : chaîne.

éléments

L'un des éléments contenus par l'élément msg. Un seul de ces éléments apparaît dans chaque message :

- trans
- rowOp
- lob
- subDeactivated
- loadDoneRcvd
- heartbeat
- errorRpt
- subSchema

Exemple :

L'exemple suivant présente un élément de message.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqcap.xsd"
      version="1.0.0" dbName="DB1">
```

éléments

`</msg>`

éléments représente l'un des éléments suivants : trans, rowOp, lob, subDeactivated, loadDoneRcvd, heartbeat, errorRpt ou subSchema.

Message de transaction

Un message de transaction contient une ou plusieurs opérations d'insertion, de mise à jour ou de suppression de ligne sur la table source. Le message de transaction contient également des informations sur l'heure à laquelle la transaction a été validée dans la base de données source, ainsi qu'un numéro d'ordre de journal fondé sur l'horaire.

Si un message de transaction dépasse la taille maximale des messages définie pour la file d'attente d'envoi, le programme Q Capture peut le diviser en plusieurs messages de transaction de plus petite taille. Chaque message d'une transaction divisée est numéroté à l'aide de l'attribut de numéro de segment de l'élément de transaction (trans). Tous les messages d'une transaction divisée portent les mêmes valeurs pour l'heure et le numéro d'ordre logique de validation.

Dans un message de transaction, l'élément trans contient une hiérarchie d'autres éléments qui décrivent le type d'opération de ligne, les attributs de chaque colonne, le type de données de la valeur de colonne et la valeur elle-même. Les sections suivantes décrivent les éléments contenus dans l'élément trans.

- «Élément de transaction (trans)»
- «Éléments d'opération de ligne (insertRow, updateRow et deleteRow)», à la page 573
- «Élément de colonne (col)», à la page 574
- «Éléments pour une valeur de colonne simple», à la page 576
- «Éléments pour une valeur de colonne double», à la page 578
- «Éléments valeur avant et valeur après (beforeVal et afterVal)», à la page 579

Élément de transaction (trans)

L'élément de transaction (trans) est contenu dans l'élément msg et contient lui-même l'un des trois éléments d'opération de ligne (insertRow, updateRow ou deleteRow).

Le tableau 100 décrit l'élément trans.

Tableau 100. Description d'élément pour trans

Nom	Propriétés
trans	Contenu complexe de type complexe non vide

Structure

```
<trans isLast="indicateur_dernier" segmentNum="numéro_segment"
  cmitLSN="numéro_ordre_logique_validation" cmitTime="heure_validation"
  authID="ID_autorisation" correlationID="ID_corrélation"
  planName="nom_plan">
```

éléments

</trans>

Détails

indicateur_dernier

Valeur booléenne qui indique si le message de transaction est le dernier message d'une transaction de base de données. Si c'est le cas, la valeur est 1 (true). Dans le cas contraire, la valeur est 0 (false). Type de données XML : booléen.

Si une transaction de base de données contient des opérations de ligne avec des colonnes LOB et que des valeurs LOB doivent être publiées, ces valeurs sont envoyées dans des messages LOB distincts après le dernier message de transaction. Dans ce cas, le dernier message d'une transaction de base de données n'est pas le dernier message de transaction, mais un message LOB.

numéro_segment

Entier positif qui indique le numéro de segment du message dans un message de transaction divisé. Type de données XML : positiveInteger.

numéro_ordre_logique_validation

Numéro d'ordre logique de validation (numéro d'ordre de journal fondé sur l'horaire) de l'instruction COMMIT de la transaction. Type de données XML : chaîne.


heure_validation

Horodatage de l'instruction COMMIT pour la transaction en temps moyen de Greenwich (GMT), formaté en microsecondes. Type de données XML : dateTime.

ID_autorisation

ID de l'utilisateur qui a mis à jour la table source. Type de données XML : chaîne.

ID_corrélation

 L'ID de corrélation (normalement un nom de travail) qui a exécuté la mise à jour source. Type de données XML : chaîne.

nom_plan

 Nom de plan associé à la transaction. Type de données XML : chaîne.

éléments

Chaque élément trans contient un ou plusieurs des éléments suivants :

- insertRow
- updateRow
- deleteRow

Exemple

L'exemple suivant affiche un élément de transaction qui contient un ou plusieurs des éléments d'insertion, de mise à jour ou de suppression de ligne.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqcap.xsd"
      version="1.0.0" dbName="DB1">
  <trans isLast="1" segmentNum="1" cmitLSN="0000:0000:0000:06d6:87ab">
```



```
commitTime="2003-10-31T12:12:12.000122">
```

```
insertRow, updateRow ou deleteRow
```

```
</trans></msg>
```

insertRow, *updateRow* ou *deleteRow* représente les éléments décrits à la section «Éléments d'opération de ligne (*insertRow*, *updateRow* et *deleteRow*)».

Éléments d'opération de ligne (*insertRow*, *updateRow* et *deleteRow*)

Dans un élément de transaction, les éléments d'opération de ligne (*insertRow*, *updateRow* et *deleteRow*) décrivent le type d'opération SQL qui est effectué sur une ligne de la table source. Chacun de ces éléments contient un ou plusieurs éléments de colonne (*col*) qui décrivent les modifications apportées aux colonnes abonnées.

Le tableau 101 décrit les éléments *insertRow*, *deleteRow* et *updateRow*.

Tableau 101. Description d'élément pour *insertRow*, *deleteRow* et *updateRow*

Nom	Propriétés
<i>insertRow</i>	Contenu complexe de type complexe non vide
<i>deleteRow</i>	Contenu complexe de type complexe non vide
<i>updateRow</i>	Contenu complexe de type complexe non vide

Structure

```
<insertRow subName="nom_publication" srcOwner="propriétaire_source"
  srcName="nom_source" rowNum="numéro_ligne" hasLOBCols="indicateur_objet_LOB">
```

```
  éléments
```

```
</insertRow>
```

```
<deleteRow subName="nom_publication" srcOwner="propriétaire_source"
  srcName="nom_source" rowNum="numéro_ligne" hasLOBCols="indicateur_objet_LOB">
```

```
  éléments
```

```
</deleteRow>
```

```
<updateRow subName="nom_publication" srcOwner="propriétaire_source"
  srcName="nom_source" rowNum="numéro_ligne" hasLOBCols="indicateur_objet_LOB">
```

```
  éléments
```

```
</updateRow>
```

Détails

nom_publication

Nom de la publication à laquelle s'applique cette opération de ligne. Type de données XML : chaîne.

propriétaire_source

Schéma de la table source de laquelle provient l'opération de ligne. Type de données XML : chaîne.

nom_source

Nom de la table source. Type de données XML : chaîne.

numéro_ligne

Si une opération de ligne comprend des colonnes d'objets LOB, cet attribut est généré pour identifier le numéro de position de l'opération de ligne dans la transaction de base de données. Cet attribut n'a pas de valeur par défaut. Type de données XML : `positiveInteger`.

indicateur_objet_LOB

Valeur booléenne qui indique si l'opération de ligne contient des colonnes LOB. Si c'est le cas, la valeur est 1 (true). La valeur par défaut est 0 (false). Type de données XML : booléen.

éléments

Un ou plusieurs éléments de colonne (col) contenus dans l'élément `insertRow`, `updateRow` ou `deleteRow`.

Exemple

L'exemple suivant présente des éléments `insertRow`, `updateRow` et `deleteRow` dans un message de transaction.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
      dbName="DB1">
  <trans isLast="1" segmentNum="1" cmitLSN="0000:0000::0000:06d6:87ab"
        cmitTime="2003-10-31T12:12:12.000122">
    <insertRow subName="S1" srcOwner="USER1" srcName="T1">
      élément_colonne
    </insertRow>
    <deleteRow subName="S1" srcOwner="USER1" srcName="T1">
      élément_colonne
    </deleteRow>
    <updateRow subName="S1" srcOwner="USER1" srcName="T1">
      élément_colonne
    </updateRow>
  </trans>
</msg>
```

élément_colonne représente l'élément de colonne décrit à la section «Élément de colonne (col)».

Élément de colonne (col)

L'élément de colonne (col) décrit le nom d'une colonne abonnée dans la table source et indique également si la colonne fait ou non partie de la clé à utiliser pour la publication. Un élément col dans une opération d'insertion ou de suppression contient une seule valeur. Dans une opération de mise à jour, l'élément col peut contenir une valeur avant et une valeur après, en fonction des options d'envoi de données que vous avez définies pour la publication.

Si vous avez spécifié que la mappe de file d'attente des publications doit envoyer des données erronées en cas d'échec de conversion de la page, les attributs `invalidData` et `rawData` sont définis sur 1 pour les éléments `col` dans les opérations d'insertion. Pour les opérations de mise à jour, les attributs `invalidData` et `rawData` sont définis sur 1 pour les éléments `beforeVal` ou `afterVal`.

Le tableau 102 décrit l'élément `col`.

Tableau 102. Description d'élément pour `col`

Nom	Propriétés
<code>col</code>	Contenu complexe de type complexe non vide

Structure

```
<col name="nom_colonne" isKey="indicateur_clé" options_données_incorrectes>
    valeur_colonne_simple_ou_double
</col>
```

Détails

nom_colonne

Nom d'une colonne abonnée dans la table source. Type de données XML : chaîne.

indicateur_clé

Facultatif : valeur booléenne indiquant si la colonne fait ou non partie de la clé à utiliser pour la publication. La valeur par défaut est 0 (false). S'il s'agit d'une colonne de clé, la valeur est 1 (true). Type de données XML : booléen.

options_données_incorrectes

Facultatif : si vous spécifiez que la mappe de files pour la publication doit publier des données lorsque des erreurs de conversion de page de code se produisent, les données sont publiées au format hexadécimal. Pour les opérations d'insertion, les attributs `invalidData` et `rawData` sont définies sur 1 (true) pour l'élément `col`. Pour les opérations de mise à jour, les attributs `invalidData` et `rawData` sont définis sur 1 (true) pour les éléments `beforeVal` ou `afterVal`. La valeur par défaut est 0 (false). Type de données XML : booléen.

valeur_colonne_simple_ou_double

Si l'élément de colonne fait partie d'une opération d'insertion ou de suppression dans la table source, il contient l'un des éléments de valeur de colonne simple. Pour les opérations de mise à jour, l'élément de colonne peut contenir une valeur de colonne double, qui contient une valeur avant et une valeur après.

Exemple

L'exemple suivant présente une opération d'insertion qui contient des valeurs de colonne simple, ainsi qu'une opération de mise à jour qui contient des valeurs de colonne double.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
    dbName="DB1">
    <trans isLast="1" segmentNum="1" cmitLSN="0000:0000::0000:06d6:87ab"
        cmitTime="2003-10-31T12:12:12.000122">
        <insertRow subName="S1" srcOwner="USER1" srcName="T1">
```

```

        <col name="COL1" isKey="1">
            valeur_colonne_simple
        </col>
        <col name="COL2">
            valeur_colonne_simple
        </col>
    </insertRow>
    <updateRow subName="S1" srcOwner="USER1" srcName="T1">
        <col name="COL1" isKey="1">
            valeur_colonne_double
        </col>
        <col name="COL2">
            valeur_colonne_double
        </col>
    </updateRow> </trans>
</msg>

```

valeur_colonne_simple représente les éléments décrits à la section «Eléments pour une valeur de colonne simple» et *valeur_colonne_double*, les éléments décrits à la section «Eléments pour une valeur de colonne double», à la page 578.

Eléments pour une valeur de colonne simple

Un élément de valeur de colonne simple contient une valeur réelle provenant de la table source. Le programme Q Capture utilise des éléments de valeur de colonne simple pour les opérations d'insertion et de suppression. Ces éléments ont pour nom le type de données de la colonne source et ne contiennent pas d'autres éléments. Si la valeur de la table source est NULL, l'élément est vide et la valeur 1 (true) est affectée à l'attribut `xsi:nil`.

Le tableau 103 décrit l'élément de valeur de colonne simple. Tous les éléments sont de type complexe et ont un contenu simple. Les éléments blob, clob et dbclob, qui comportent des données LOB, sont toujours vides parce que les données d'un objet LOB sont envoyées dans un message LOB distinct. Les éléments blob, clob et dbclob n'ont pas l'attribut `xsi:nil`.

Tableau 103. Descriptions d'élément pour la valeur de colonne simple

Nom	type de données XML	Format de données de la valeur
smallint	court	
integer	integer	
bigint	long	
float	flottant (32 bits)	[-]d.dddde[- +]dd [-]d.dddde[- +]dd
	double (64 bits)	
real	float	
double	double	
decimal	decimal	
date	date	AAAA-MM-JJ
time	time	HH:MM:SS.SSS

Tableau 103. Descriptions d'élément pour la valeur de colonne simple (suite)

Nom	type de données XML	Format de données de la valeur
timestamp	date/heure	AAAA-MM-JJTHH:MM:SS.SSS
char	chaîne	
varchar	chaîne	
long varchar	chaîne	
bitchar	binaire hexadécimal	
bitvarchar	binaire hexadécimal	
bitlongvarchar	binaire hexadécimal	
graphic	chaîne	
vargraphic	chaîne	
longvargraphic	chaîne	
rowid	binaire hexadécimal	
blob	binaire hexadécimal	
clob	chaîne	
dbclob	chaîne	

Structure

```
<type_données xsi:nil="indicateur_null">valeur</type_données>
```

Détails

type_données

Type de données de la colonne dans la table source. Ce type de données est utilisé pour nommer l'élément.

indicateur_Null

Facultatif : entier qui indique si la colonne source contient une valeur NULL. La valeur par défaut est 0 (false). Si la colonne source contient une valeur NULL, la valeur de cet attribut est 1 (true). Les éléments blob, clob et dbclob n'ont pas cet attribut. Type de données XML : booléen.

valeur

Valeur réelle dans la colonne source. Si la valeur source est NULL ou une valeur LOB, l'élément est vide.

Exemple

L'exemple suivant présente une opération d'insertion comportant des valeurs de colonne simple de 222 dans une colonne de clé avec des données de type entier et la chaîne Hello dans une colonne sans clé avec des données de type varchar. Cet exemple illustre également une opération de suppression de la ligne avec une valeur de colonne simple de 222 dans une colonne de clé avec des données de type entier.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
  dbName="DB1">
  <trans isLast="1" segmentNum="1" cmitLSN="0000:0000::0000:06d6:87ab"
    cmitTime="2003-10-31T12:12:12.000122">
    <insertRow subName="S1" srcOwner="USER1" srcName="T1">
      <col name="COL1" isKey="1">
```

```

        <integer>222</integer>
    </col>
    <col name="COL2">
        <varchar>Hello</varchar>
    </col>
</insertRow>
<deleteRow subName="S1" srcOwner="USER1" srcName="T1">
    <col name="COL1" isKey="1">
        <integer>222</integer>
    </col>
</deleteRow>
</trans>
</msg>

```

Éléments pour une valeur de colonne double

Les éléments de valeur de colonne double sont utilisés dans les opérations de mise à jour lorsque le programme Q Capture doit envoyer des valeurs avant et après à partir des colonnes source. Dans les messages, le programme Q Capture envoie les valeurs avant des colonnes de clé qui ont changé. Il envoie les valeurs avant des colonnes sans clé qui ont changé si l'option d'envoi des données BEFORE_VALUES est définie sur "Yes" pour la publication. Si les valeurs avant et après sont identiques, seul l'élément de valeur après (afterValue) est utilisé.

Tous les éléments de valeur de colonne double à l'exception de blob, clob et dbclob ne sont pas vides, sont de type complexe et ont un contenu complexe. Les éléments blob, clob et dbclob sont toujours vides et n'admettent pas la valeur Null. Les éléments de valeur de colonne double n'ont pas d'attribut. Pour une description des éléments de valeur de colonne double, voir «Éléments pour une valeur de colonne simple», à la page 576.

Structure

```

<type_données>
    éléments
</type_données>

```

Détails

type_données

Type de données de la colonne dans la table source. Ce type de données est utilisé pour nommer l'élément. Si le type de données est blob, clob ou dbclob, les éléments sont vides et n'admettent pas la valeur Null.

éléments

L'un des éléments beforeValue ou afterValue, ou les deux, ou vide pour blob, clob et dbclob.

Exemple

L'exemple suivant présente les éléments de valeur de colonne double pour :

- une colonne de clé (type de données entier) qui a été modifiée ;
- une colonne sans clé (type de données varchar) qui a été modifiée, mais l'option d'envoi des données BEFORE_VALUES pour la publication a été définie sur "No."

```

<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"

```

```

dbName="DB1">
  <trans isLast="1" segmentNum="1" cmitLSN="0000:0000::0000:06d6:87ab"
    cmitTime="2003-10-31T12:12:12.000122">
    <updateRow subName="S1" srcOwner="USER1" srcName="T1">
      <col name="COL1" isKey="1">
        <integer>
          valeuravant
          valeuraprès
        </integer>
      </col>
      <col name="COL2">
        <varchar>
          valeuraprès
        </varchar>
      </col>
    </updateRow>
  </trans>
</msg>

```

valeuravant et *valeuraprès* représentent les éléments décrits à la section «Eléments valeur avant et valeur après (beforeVal et afterVal)».

Eléments valeur avant et valeur après (beforeVal et afterVal)

Les éléments valeur avant et valeur après (beforeVal et afterVal) contiennent des valeurs réelles issues de la table source. Ces éléments sont utilisés dans les opérations de mise à jour des colonnes de clé qui ont été modifiée et des colonnes sans clé modifiées lorsque l'option d'envoi des données BEFORE_VALUES pour la publication est définie sur "Yes." Si la publication demande l'envoi de valeurs avant et que la valeur de la colonne source n'a pas changé, seul l'élément afterVal est utilisé. Si la valeur de la table source est NULL, les éléments sont vides et la valeur 1 (true) est affectée à l'attribut xsi:null.

Le tableau 104 décrit les éléments beforeVal et afterVal.

Tableau 104. Descriptions d'élément pour beforeVal et afterVal

Nom	Propriétés
beforeVal	Contenu simple de type complexe n'admettant pas la valeur Null
afterVal	Contenu simple de type complexe n'admettant pas la valeur Null, facultatif

Structure

```

<beforeVal xsi:nil="indicateur_null" invalidData="indicateur_incorrect"
  rawData="indicateur_données_brutes">valeur</beforeVal>
<afterVal xsi:nil="indicateur_null" invalidData="indicateur_incorrect"
  rawData="indicateur_données_brutes">indicateur</afterVal>

```

Détails

indicateur_Null

Facultatif : entier qui indique si la valeur de la colonne source est NULL. La valeur par défaut est 0 (false). Si la colonne source contient une valeur NULL, la valeur de cet attribut est 1 (true). Type de données XML : booléen.

valeur

Valeur réelle dans la colonne source. Si la valeur source est NULL, l'élément est vide.

indicateur_incorrect

Facultatif : si vous spécifiez que la mappe de files pour la publication doit publier des données lorsque des erreurs de conversion de page de code se produisent, les données sont publiées au format hexadécimal. Si la valeur avant ne peut pas être convertie, les attributs `invalidData` et `rawData` sont définis sur 1 (true) pour l'élément `beforeVal`. La valeur par défaut est 0 (false).
Type de données XML : booléen.

indicateur_données_brutes

Facultatif : si vous spécifiez que la mappe de files pour la publication doit publier des données lorsque des erreurs de conversion de page de code se produisent, les données sont publiées au format hexadécimal. Si la valeur après ne peut pas être convertie, les attributs `invalidData` et `rawData` sont définis sur 1 (true) pour l'élément `afterVal`. La valeur par défaut est 0 (false). Type de données XML : booléen.

Exemple

L'exemple suivant présente une opération de mise à jour dans laquelle la valeur de la colonne de clé de 222 n'a pas changé (seul l'élément `afterVal` est utilisé) et dans laquelle une colonne varchar de la même ligne a changé de "Hello" à NULL. Dans ce cas, l'option `BEFORE_VALUES` de la publication est définie sur "Yes."

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
      dbName="DB1">
  <trans isLast="1" segmentNum="1" cmitLSN="0000:0000::0000:06d6:87ab"
        cmitTime="2003-10-31T12:12:12.000122">
    <updateRow subName="S1" srcOwner="USER1" srcName="T1">
      <col name="COL1" isKey="1">
        <integer>
          <afterVal>222</afterVal>
        </integer>
      </col>
      <col name="COL2">
        <varchar>
          <beforeVal>Hello</beforeVal>
          <afterVal xsi:nil="1"/>
        </varchar>
      </col>
    </updateRow>
  </trans>
</msg>
```

Message d'opération de ligne

Un message d'opération de ligne contient une opération d'insertion, de mise à jour ou de suppression de ligne sur la table source. Dans un message d'opération de ligne, l'élément de message (`msg`) contient un élément d'opération de ligne (`rowOp`).

Un message d'opération de ligne ne doit pas dépasser la taille de message maximale définie pour la file d'attente d'envoi. Les messages d'opération de ligne qui dépassent cette taille ne peuvent pas être divisés en plusieurs messages. Dans les messages d'opération de ligne, les insertions, mises à jour ou suppression qui dépendent d'une transaction ont la même heure et le même numéro d'ordre logique de validation. Si les messages LOB suivent le message opération de ligne,

ce dernier contient un attribut pour indiquer qu'il n'est pas le dernier message de l'opération de ligne provenant de la base de données source.

Le tableau 105 décrit l'élément rowOp.

Tableau 105. Description d'élément pour rowOp

Nom	Propriétés
rowOp	Contenu complexe de type complexe non vide

Structure

```
<rowOp cmitLSN="numéro_ordre_logique_validation"
  cmitTime="heure_validation" isLast="indicateur_dernier"
  authID="ID_autorisation" correlationID="ID_corrélation"
  planName="nom_plan">
```

éléments

```
</rowOp>
```

Détails

numéro_ordre_logique_validation

Numéro d'ordre logique de validation (numéro d'ordre de journal fondé sur l'horaire) de l'instruction COMMIT de la transaction. Type de données XML : chaîne.

heure_validation

Horodatage de l'instruction COMMIT pour la transaction en temps moyen de Greenwich (GMT), formaté en microsecondes. Type de données XML : dateTime.

indicateur_dernier

Facultatif : valeur booléenne qui indique si le message d'opération de ligne est le dernier message d'une opération de ligne de la base de données source. Si des messages LOB suivent le message d'opération de ligne, la valeur est définie à 0 (false). Cet attribut n'a pas de valeur par défaut. Type de données XML : booléen.

ID_autorisation

ID de l'utilisateur qui a mis à jour la table source. Type de données XML : chaîne.

ID_corrélation

z/OS uniquement : ID de corrélation (en principe, un nom de travail) ayant exécuté la mise à jour source. Type de données XML : chaîne.

nom_plan

z/OS uniquement : Nom de plan associé à la transaction à laquelle la ligne appartient. Type de données XML : chaîne.

éléments

Chaque élément rowOp contient l'un des éléments suivants :

- insertRow
- updateRow
- deleteRow

Exemple

L'exemple suivant présente un élément d'opération de ligne qui contient un élément `insertRow`, `updateRow` ou `deleteRow`.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0" dbName="DB1">
  <rowOp cmitLSN="0000:0000::0000:06d6:87ab"
        cmitTime="2003-10-31T12:12:12.000122">
    insertRow, deleteRow ou updateRow
  </rowOp></msg>
```

Où *insertRow*, *updateRow* ou *deleteRow* représente les éléments décrits dans la section «Message de transaction», à la page 571.

Message d'objet LOB

Un message d'objet LOB transmet tout ou partie des données d'une colonne de la table source qui contient une valeur d'objet LOB : BLOB (binary large object), CLOB (character large object) ou DBCLOB (double-byte character large object).

Chaque message LOB contient des données provenant d'au moins une valeur LOB dans la table source. Le programme Q Capture peut diviser une valeur LOB en plusieurs messages LOB si la valeur dépasse la taille de la mémoire tampon des messages LOB déterminée par le programme Q Capture. La taille de la mémoire tampon peut aller jusqu'à la taille de message maximale définie pour la file d'attente d'envoi. Tous les messages qui contiennent une partie de la même valeur LOB ont le même nom de publication, propriétaire de table source, nom de table source, numéro de ligne et nom de colonne.

Les messages qui contiennent des valeurs LOB sont envoyés après les messages qui contiennent la transaction ou l'opération de ligne dont dépendent les valeurs LOB. L'attribut `isLast` dénote le dernier message d'une valeur LOB divisée, qui est aussi le dernier message d'une transaction ou opération de ligne.

Dans un message LOB, l'élément LOB (`lob`) est contenu dans l'élément de message (`msg`) et contient lui-même un élément unique de valeur de colonne LOB.

Le tableau 106 décrit l'élément `lob`.

Tableau 106. Description d'élément pour `lob`

Nom	Propriétés
<code>lob</code>	Contenu complexe de type complexe non vide

Structure

```
<lob isLast="indicateur_dernier" subName="nom_publication"
     srcOwner="proprietaire_source" srcName="nom_source" rowNum="numero_ligne"
     colName="nom_colonne" totalDataLen="longueur_donnees_objet_LOB"
     dataLen="longueur_donnees_segment">
    valeur_colonne_objet_LOB
</lob>
```

Détails

indicateur_dernier

Valeur booléenne qui indique s'il s'agit du dernier message d'une transaction ou d'une opération de ligne. S'il s'agit du dernier message, la valeur est 1 (true). Dans le cas contraire, la valeur est 0 (false). Type de données XML : booléen.

nom_publication

Nom de la publication XML qui comprend la valeur LOB. Type de données XML : chaîne.

proprietaire_source

Schéma de la table source de laquelle provient l'objet LOB. Type de données XML : chaîne.

nom_source

Nom de la table source. Type de données XML : chaîne.

numero_ligne

Dans la transaction de base de données, numéro de position de l'opération de ligne qui contient la valeur LOB. Type de données XML : positiveInteger.

nom_colonne

Nom de la colonne dans la table source qui contient la valeur LOB. Type de données XML : chaîne.

longueur_donnees_objet_LOB

Longueur totale de la valeur LOB contenue dans la table source, en octets. Type de données XML : nonNegativeInteger.

longueur_donnees_segment

Longueur des données LOB contenues dans un segment de message unique, en octets. Type de données XML : nonNegativeInteger.

valeur_colonne_objet_LOB

L'un des trois éléments de valeur de colonne LOB qui décrit le type de données de la valeur LOB. Les trois éléments sont blob, clob et dbclob.

Exemple

L'exemple suivant présente un message LOB.

```
<xml version="1.0" encoding="UTF-8" ?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="mqcap.xsd"
  version="1.0.0" dbName="DB1">
  <lob isLast="0" subName="S1" srcOwner="USER1" srcName="T1" rowNum="3"
    colName="LOBCOL" totalDataLen="92675" dataLen="100">
    valeur_colonne_objet_LOB
  </lob>
</msg>
```

valeur_colonne_objet_LOB décrit l'un des trois éléments décrits à la section «Valeur de colonne LOB».

Valeur de colonne LOB

Les trois éléments de valeur de colonne LOB contiennent chacun des données LOB réelles provenant de la table source. Les éléments sont nommés d'après leur type

de données, blob, clob ou dbclob. Si la valeur de la table source est NULL, les éléments sont vides et la valeur 1 (true) est affectée à l'attribut xsi:nil.

Le tableau 107 décrit les éléments de valeur de colonne LOB.

Tableau 107. Description d'élément pour les valeurs de colonne LOB

Nom	Propriétés
blob	Contenu simple de type complexe n'admettant pas la valeur Null
clob	Contenu simple de type complexe n'admettant pas la valeur Null
dbclob	Contenu simple de type complexe n'admettant pas la valeur Null

Structure

```
<type_données xsi:nil="indicateur_Null">  
  valeur_objet_LOB  
</type_données>
```

Détails

type_données

Type de données de la colonne dans la table source. Ce type de données est utilisé pour nommer l'élément.

indicateur_Null

Facultatif : valeur booléenne indiquant si la valeur de la colonne source est NULL. La valeur par défaut est 0 (false). Si la colonne source contient une valeur NULL, la valeur de cet attribut est 1 (true). Type de données XML : booléen.

valeur_objet_LOB

Données réelles de l'objet LOB dans la table source.

Exemple

L'exemple suivant présente un message LOB qui comprend 100 octets du total de 92 675 octets de données d'une valeur d'objet CLOB.

```
<xml version="1.0" encoding="UTF-8" ?>  
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  xsi:noNamespaceSchemaLocation="mqcap.xsd"  
  version="1.0.0" dbName="DB1">  
  <lob isLast="0" subName="S1" srcOwner="USER1" srcName="T1" rowNum="3"  
    colName="LOBCOL" totalDataLen="92675" dataLen="100">  
    <clob>données_objet_LOB</clob>  
  </lob>  
</msg>
```

Message d'abonnement désactivé

Un message d'abonnement désactivé confirme que le programme Q Capture a reçu le message de désactivation d'abonnement de l'application utilisateur.

Dans un message d'abonnement désactivé, l'élément de message (msg) contient un élément d'abonnement désactivé (subDeactivated).

Le tableau 108 décrit l'élément subDeactivated.

Tableau 108. Description d'élément pour subDeactivated

Nom	Propriétés
subDeactivated	Type complexe vide

Structure

```
<subDeactivated subName="nom_publication" srcOwner="propriétaire_source"
  srcName="nom_source" stateInfo="information_état"/>
```

Détails

nom_publication

Nom de la publication qui a été désactivée. Type de données XML : chaîne.

propriétaire_source

Schéma de la table source pour la publication. Type de données XML : chaîne.

nom_source

Nom de la table source. Type de données XML : chaîne.

information_état

Informations complémentaires sur l'état de la publication. Cet attribut contient un numéro de message ASN. Type de données XML : chaîne.

Exemple

L'exemple suivant présente un message d'abonnement désactivé.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
  dbName="DB1">

  <subDeactivated subName="S1" srcOwner="USER1" srcName="T1"
    stateInfo="ASN7019I"/>

</msg>
```

Message de chargement terminé reçu

Le message de chargement terminé reçu confirme que le programme Q Capture a reçu le message de chargement terminé de l'application utilisateur. Le message de chargement terminé signifie qu'une table cible est chargée.

Dans un message de chargement terminé reçu, l'élément de message (msg) contient un élément de chargement terminé reçu (loadDoneRcvd).

Le tableau 109 décrit l'élément loadDoneRcvd.

Tableau 109. Description d'élément pour loadDoneRcvd

Nom	Propriétés
loadDoneRcvd	Type complexe vide

Structure

```
<loadDoneRcvd subName="nom_publication" srcOwner="propriétaire_source"
  srcName="nom_source" stateInfo="information_état"/>
```

Détails

nom_publication

Nom de la publication pour laquelle la table cible a été chargée. Type de données XML : chaîne.

propriétaire_source

Schéma de la table source pour la publication. Type de données XML : chaîne.

nom_source

Nom de la table source. Type de données XML : chaîne.

information_état

Informations complémentaires sur l'état de la publication. Cet attribut contient un numéro de message ASN. Type de données XML : chaîne.

Exemple

L'exemple suivant présente un message de chargement terminé reçu.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
  dbName="DB1">

  <loadDoneRcvd subName="S1" srcOwner="USER1" srcName="T1"
    stateInfo="ASN7019I"/>

</msg>
```

Message de rapport d'erreurs

Le programme Q Capture envoie un message de rapport d'erreurs lorsqu'il ne peut pas accéder à la demande qu'une application utilisateur a effectuée par le biais d'un message de contrôle.

Par exemple, le programme Q Capture envoie un message de rapport d'erreurs s'il ne parvient pas à activer ou à désactiver une publication ou à accuser réception d'un message chargement terminé. Le programme Q Capture consigne également ces erreurs dans son journal. S'il ne peut pas envoyer de rapport d'erreurs parce que la file d'attente d'envoi n'est pas disponible, le programme Q Capture consigne quand même l'erreur dans son journal. Les erreurs liées à WebSphere MQ n'entraînent pas la génération de messages de rapport d'erreurs.

Dans un message de rapport d'erreurs, l'élément de message (msg) contient un élément de rapport d'erreurs (errorRpt).

Le tableau 110 décrit l'élément de rapport d'erreurs.

Tableau 110. Description d'élément pour errorRpt

Nom	Propriétés
errorRpt	Type complexe vide

Structure

```
<errorRpt subName="nom_publication" srcOwner="propriétaire_source"
  srcName="nom_source" errorMsg="texte_message"/>
```

Détails

nom_publication

Nom de la publication qui a généré une erreur. Type de données XML : chaîne.

propriétaire_source

Schéma de la table source pour la publication. Type de données XML : chaîne.

nom_source

Nom de la table source. Type de données XML : chaîne.

texte_message

Texte du message d'erreur. Type de données XML : chaîne.

Exemple

L'exemple suivant présente un message de rapport d'erreurs généré lorsque le programme Q Capture n'a pas réussi à activer une publication

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
  dbName="DB1">

  <errorRpt subName="S1" srcOwner="USER1" srcName="T1"
    errorMsg="texte_message"/>

</msg>
```

Où *texte_message* représente le texte du message d'erreur.

Message de pulsation

Un message de pulsation indique à l'application utilisateur qu'un programme Q Capture est encore en cours d'exécution. Le programme Q Capture place ce type de message dans les files d'attente d'envoi actives chaque fois que l'intervalle de pulsation de la mappe de files pour la publication qui contient la file d'attente d'envoi est écoulé s'il n'y a aucun message à placer dans la file. Si le programme Q Capture parvient à la fin du journal avant que cet intervalle ne soit écoulé, il envoie un message de pulsation sans aucune information sur l'heure de la dernière validation.

Dans un message de pulsation, l'élément de message (msg) contient un élément de pulsation (heartbeat).

Le tableau 111 décrit l'élément de pulsation.

Tableau 111. Description d'élément pour heartbeat

Nom	Propriétés
heartbeat	Type complexe vide

Structure

```
<heartbeat sendQName="nom_file_attente_envoi"
  lastCmitTime="heure_dernière_validation"/>
```

Détails

nom_file_attente_envoi

Nom de la file d'attente d'envoi dans laquelle le programme Q Capture a placé le message de pulsation. Type de données XML : chaîne.

heure_dernière_validation

Facultatif : horodatage de la dernière transaction validée en temps moyen de Greenwich (GMT). Cet attribut est facultatif et n'a pas de valeur par défaut. Type de données XML : dateTime.

Exemple

L'exemple suivant présente un message de pulsation.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
      dbName="DB1">

      <heartbeat sendQName="Q1" lastcmitTime="2003-10-31T12:12:12.000122"/>

</msg>
```

Message de schéma d'abonnement (subSchema)

Le programme Q Capture envoie un message de schéma d'abonnement pour confirmer qu'il a activé ou réinitialisé une publication. Ce message comporte des informations sur la publication, dont les noms de la table source et de la file d'attente d'envoi, les options d'envoi de données et des informations sur la phase de chargement. Le message de schéma d'abonnement est envoyé en réponse à un message d'activation d'abonnement, à une commande **reinit** ou à un signal REINIT_SUB.

Dans un message de schéma d'abonnement, l'élément de message (msg) contient un élément de schéma d'abonnement (subSchema), qui contient à son tour un ou plusieurs éléments de colonne (col). Les sections suivantes décrivent les deux éléments.

- «Élément de schéma d'abonnement (subSchema)»
- «Élément de colonne (col) dans un message de schéma d'abonnement», à la page 590

Élément de schéma d'abonnement (subSchema)

Par le biais de ses attributs, l'élément subSchema fournit des détails sur une publication. L'élément subSchema contient un ou plusieurs éléments de colonne (col).

Le tableau 112 décrit l'élément subSchema.

Tableau 112. Description d'élément pour subSchema

Nom	Propriétés
subSchema	Contenu simple de type complexe non vide

Structure

```
<subSchema subname="nom_publication"
            srcOwner="propriétaire_source"
            srcName="nom_source">
```



```

sendQName="nom_file_attente_envoi"
allChangedRows="option_ALL_CHANGED_ROWS"
beforeValues="option_BEFORE_VALUES"
changedColsOnly="option_CHANGED_COLS_ONLY"
hasLoadPhase="option_phase_chargement"
dbServerType="système_exploitation"
dbRelease="niveau_édition_DB2"
dbInstance="nom_instance_DB2"
capRelease="niveau_édition_Q_capture">

```

éléments_colonne

</subSchema>

Détails

nom_publication

Nom de la publication qui a été activée ou réinitialisée. Type de données XML : chaîne.

propriétaire_source

Schéma de la table source pour la publication. Type de données XML : chaîne.

nom_source

Nom de la table source. Type de données XML : chaîne.

nom_file_attente_envoi

Nom de la file d'attente d'envoi qui est spécifiée pour la publication. Type de données XML : chaîne.

option_ALL_CHANGED_ROWS

Facultatif : valeur booléenne indiquant si l'option d'envoi de données ALL_CHANGED_ROWS est spécifiée pour la publication. La valeur par défaut est 0 (false). Si l'option est spécifiée, la valeur est 1 (true). Type de données XML : booléen.

option_BEFORE_VALUES

Facultatif : valeur booléenne indiquant si l'option d'envoi de données BEFORE_VALUES est spécifiée pour la publication. La valeur par défaut est 0 (false). Si l'option est spécifiée, la valeur est 1 (true). Type de données XML : booléen.

option_CHANGED_COLS_ONLY

Facultatif : valeur booléenne indiquant si l'option d'envoi de données CHANGED_COLS_ONLY est spécifiée pour la publication. La valeur par défaut est 0 (false). Si l'option est spécifiée, la valeur est 1 (true). Type de données XML : booléen.

option_phase_chargement

Facultatif : indicateur définissant si la publication comporte une phase de chargement. La valeur par défaut est "none" pour aucune phase de chargement. Si une phase de chargement est spécifiée, la valeur est "external". Type de données XML : loadPhaseEnumType.

système_exploitation

Facultatif : système d'exploitation de la base de données ou du sous-système source. La valeur par défaut est QDB2/6000 (DB2 pour AIX). Type de données XML : dbServerTypeEnumType.

niveau_édition_DB2

Niveau d'édition DB2 de la base de données ou du sous-système source. Type de données XML : chaîne.

nom_instance_DB2

Nom de l'instance DB2 qui contient la base de données source. Type de données XML : chaîne.

niveau_édition_Q_capture

Niveau d'édition du programme Q Capture. Type de données XML : chaîne.

éléments_colonne

Un ou plusieurs éléments de colonne (col) qui comportent des informations sur chaque colonne de la table source.

Le tableau 113 fournit des détails complémentaires sur deux types de données XML utilisés dans les attributs pour l'élément subSchema.

Tableau 113. Autres descriptions de types de données pour les attributs subSchema

Nom de type	Type de base	Valeurs
loadPhaseEnumType	chaîne	none, external
dbServerTypeEnumType	chaîne	QDB2, QDB2/6000, QDB2/HPUX, QDB2/NT, QDB2/SUN, QDB2/LINUX, QDB2/Windows
		Remarque : QDB2 seul implique DB2 pour z/OS.

Exemple

Le message de schéma d'abonnement suivant serait envoyé pour une publication spécifiant l'option d'envoi de données BEFORE_VALUES et une phase de chargement.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
  dbName="DB1">
  <subSchema subname="S1"
    srcOwner="USER1"
    srcName="T1"
    sendQName="Q1"
    beforeValues="yes"
    hasLoadPhase="external"
    dbServerType="QDB2/6000"
    dbRelease="8.2.0"
    dbInstance="DB2INST"
    capRelease="8.2.0">
```

élément_colonne

```
</subSchema>
</msg>
```

élément_colonne représente un ou plusieurs éléments de colonne, qui sont décrits à la section «Élément de colonne (col) dans un message de schéma d'abonnement».

Élément de colonne (col) dans un message de schéma d'abonnement

Dans le message de schéma d'abonnement, l'élément de colonne (col) comporte des informations sur chaque colonne de la table source.

Le tableau 114 décrit l'élément col dans un message de schéma.

Tableau 114. Description d'élément pour col

Nom	Propriétés
col	Type complexe vide

Structure

```
<col name="nom_colonne"  
      type="type_données"  
      len="longueur_données"  
      precision="précision_données"  
      scale="échelle_décimale"  
      codepage="numéro_page_codes"  
      isKey="indicateur_clé"/>
```

Détails

nom_colonne

Nom de la colonne dans la table source. Type de données XML : chaîne.

type_données

Type de données de la colonne source. Il doit s'agir de l'un des types de données définis pour le type de données XML `dataTypeEnumType`. Pour la liste, reportez-vous au tableau 115. Type de données XML : `dataTypeEnumType`.

Tableau 115. Autre description de type de données pour `dataTypeEnumType`

Nom de type	Type de base	Valeurs
<code>dataTypeEnumType</code>	chaîne	smallint, integer, bigint, float, real, double, decimal, char, varchar, longvarchar, bitchar, bitvarchar, bitlongvarchar, graphic, vargraphic, longvargraphic, time, timestamp, date, rowid, blob, clob, dbclob

données_données

Facultatif : longueur maximale des données dans la colonne source. Type de données XML : `unsignedInt`.

précision_données

Facultatif : pour les types de données décimales, la précision du nombre. Type de données XML : `unsignedShort`.

échelle_décimale

Facultatif : pour les données de type décimal, l'échelle du nombre. Type de données XML : `unsignedShort`.

numéro_page_codes

Facultatif : page de codes des données de type caractères. La valeur par défaut est 0. Type de données XML : `unsignedShort`.

indicateur_clé

Facultatif : valeur booléenne indiquant s'il s'agit ou non d'une colonne de clé. La valeur par défaut est 0 (false). S'il s'agit d'une colonne de clé, la valeur est 1 (true). Type de données XML : booléen.

Exemple

L'exemple suivant présente deux éléments de colonne dans un message de schéma d'abonnement.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
      dbName="DB1">
  <subSchema subname="S1"
            srcOwner="USER1"
            srcName="T1"
            sendQName="Q1"
            beforeValues="yes"
            hasLoadPhase="external"
            dbServerType="QDB2/6000"
            dbRelease="8.2.0"
            dbInstance="DB2INST"
            capRelease="8.2.0">
    <col name="COL1" type="integer" len="4"/>
    <col name="COL2" type="varchar" len="50" codepage="1208"/>
  </subSchema>
</msg>
```

Message d'ajout de colonne

Un message d'ajout de colonne indique à l'application utilisateur qu'un programme Q Capture a ajouté une colonne à une publication existante. Ce message est envoyé en réponse à l'insertion par un utilisateur ou une application utilisateur d'un signal ADDCOL dans la table IBMQREP_SIGNAL.

Dans un message d'ajout de colonne, l'élément de message (msg) contient un élément d'ajout de colonne (addColumn). L'élément d'ajout de colonne contient un élément de schéma de colonne (col) qui fournit des informations sur la colonne ajoutée à la table source.

Le tableau 116 décrit l'élément addColumn.

Tableau 116. Description d'élément pour addColumn

Nom	Propriétés
addColumn	Contenu simple de type complexe non vide

Structure

```
<addColumn subName="nom_publication" srcOwner="propriétaire_source"
           srcName="nom_source">
  élément_colonne
</addColumn>
```

Détails

nom_publication

Nom de la publication à laquelle la colonne a été ajoutée. Type de données XML : chaîne.

propriétaire_source

Schéma de la table source pour la publication. Type de données XML : chaîne.

nom_source

Nom de la table source. Type de données XML : chaîne.

élément_colonne

Élément de schéma de colonne (col) qui contient des détails sur la colonne ajoutée, tels que le nom, le type de données, la longueur des données et s'il s'agit ou non d'une colonne de clé.

Exemple

L'exemple suivant présente un message d'ajout de colonne.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqcap.xsd" version="1.0.0"
      dbName="DB1">
  <addColumn subName="S1" srcOwner="USER1" srcName="T1">
```

élément_colonne

```
</addColumn>
</msg>
```

Où *élément_colonne* représente l'élément de colonne décrit dans la section «Élément de colonne (col) dans un message de schéma d'abonnement», à la page 590.

Structure des messages provenant d'une application utilisateur vers Q Capture

Une application utilisateur communique avec un programme Q Capture en envoyant des messages à sa file d'attente d'administration. Ces messages sont appelés messages de contrôle. L'application utilisateur se sert de ces messages pour signaler qu'une table cible est chargée ou pour demander que le programme Q Capture active ou désactive une publication ou invalide une file d'attente d'envoi.

Les rubriques suivantes décrivent la structure des messages de contrôle envoyés par une application utilisateur à un programme Q Capture.

Liste des messages provenant d'une application utilisateur vers Q Capture

Le tableau 117 décrit les quatre types de messages de contrôle envoyés par une application utilisateur à un programme Q Capture.

Tableau 117. Messages de contrôle envoyés par une application utilisateur à un programme Q Capture

Type de message	Description
Invalidation de la file d'attente d'envoi	Demande à un programme Q Capture d'invalider une file d'attente d'envoi en effectuant l'opération sur erreur de la file d'attente que vous avez indiquée.
Chargement terminé	Indique à un programme Q Capture que la table cible d'une publication est chargée.
Activation d'abonnement	Demande à un programme Q Capture d'activer une publication.
Désactivation d'abonnement	Demande à un programme Q Capture de désactiver une publication.

msg : Élément principal des messages XML provenant d'une application utilisateur vers Q Capture

L'élément msg est l'élément racine de tous les messages de contrôle envoyés par une application utilisateur à un programme Q Capture.

Le tableau 118 décrit l'élément msg.

Tableau 118. Description d'élément pour msg (d'une application utilisateur à un programme Q Capture)

Nom	Propriétés
msg	Contenu complexe de type complexe non vide

Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="instance_schéma_XML"
      xsi:noNamespaceSchemaLocation="document_schéma"
      version="version">

  elements

</msg>
```

Détails

instance_schéma_XML

URL de l'instance de schéma XML. Pour la publication d'événements, cette URL est www.w3.org/2001/XMLSchema-instance. Type de données XML : chaîne.

document_schéma

Nom de fichier du document de schéma XML. L'espace de nom XML n'est pas pris en charge dans la publication d'événements parce que les messages ne se rapportent qu'à un seul schéma XML. Les messages envoyés par une application utilisateur à un programme Q Capture se rapportent au document de schéma `mqsub.xsd`. Type de données XML : chaîne.

version

Version du schéma de message XML. Pour DB2 UDB version 8.2, il s'agit de la version 1.0.0. Type de données XML : chaîne.

éléments

L'un des éléments contenus par l'élément msg. Un seul de ces éléments apparaît dans chaque message :

- invalidateSendQ
- loadDone
- activateSub
- deactivateSub

Exemple

L'exemple suivant présente un message envoyé par une application utilisateur au programme Q Capture.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqsub.xsd" version="1.0.0">
```

éléments

</msg>

éléments représente l'un des éléments suivants : invalidateSendQ, loadDone, activateSub, deactivateSub.

Message invalidateSendQ

Une application d'abonnement envoie au programme Q Capture un message d'invalidation de la file d'attente d'envoi lorsqu'elle détecte une erreur dans une file d'attente d'envoi et demande au programme Q Capture d'effectuer l'opération sur erreur indiquée pour la publication XML.

Le tableau 119 décrit l'élément invalidateSendQ.

Tableau 119. Description d'élément pour invalidateSendQ

Nom	Propriétés
invalidateSendQ	Type complexe vide

Structure

```
<invalidateSendQ sendQName="nom_file_attente_envoi"/>
```

Détails

nom_file_attente_envoi

Nom de la file d'attente d'envoi que le programme Q Capture doit invalider.
Type de données XML : chaîne.

Exemple

L'exemple suivant présente un message d'invalidation de file d'attente d'envoi.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqsub.xsd" version="1.0.0">
  <invalidateSendQ sendQName="S1"/>
</msg>
```

Message de chargement terminé

Un message de chargement terminé indique au programme Q Capture qu'une table cible est chargée. Le programme Q Capture répond à un message de chargement terminé en envoyant un message de chargement terminé reçu.

Le tableau 120 décrit l'élément loadDone.

Tableau 120. Description d'élément pour loadDone

Nom	Propriétés
loadDone	Type complexe vide

Structure

```
<loadDone subName="nom_publication"/>
```

Détails

nom_publication

Nom de la publication qui a terminé sa phase de chargement. Type de données XML : chaîne.

Exemple

L'exemple suivant présente un message de chargement terminé.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqsub.xsd" version="1.0.0">

    <loadDone subName="S1"/>

</msg>
```

Message d'activation d'abonnement

Un message d'activation d'abonnement indique à un message Q Capture de commencer la capture des modifications pour une publication.

Le tableau 121 décrit l'élément activateSub.

Tableau 121. Description d'élément pour activateSub

Nom	Propriétés
activateSub	Type complexe vide

Structure

```
<activateSub subName="nom_publication"/>
```

Détails

nom_publication

Nom de la publication que le programme Q Capture doit activer. Type de données XML : chaîne.

Exemple

L'exemple suivant présente un message d'activation d'abonnement.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqsub.xsd" version="1.0.0">

    <activateSub subName="S1"/>

</msg>
```

Message de désactivation d'abonnement

Un message de désactivation d'abonnement indique au programme Q Capture d'arrêter la capture des modifications pour une publication.

Le tableau 122, à la page 597 décrit l'élément deactivateSub.

Tableau 122. Description d'élément pour deactivateSub

Nom	Propriétés
deactivateSub	Type complexe vide

Structure

```
<deactivateSub subName="nom_publication"/>
```

Détails

nom_publication

Nom de la publication que le programme Q Capture doit désactiver. Type de données XML : chaîne.

Exemple

L'exemple suivant présente un message de désactivation d'abonnement.

```
<?xml version="1.0" encoding="UTF-8"?>
<msg xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="mqsub.xsd" version="1.0.0">
    <deactivateSub subName="S1"/>
</msg>
```

Contacteur IBM

Vous pouvez contacter IBM pour bénéficier du service de support technique, des services de logiciels, de données sur les produits et d'informations générales. Vous pouvez également transmettre vos commentaires à IBM sur les produits et la documentation.

Le tableau suivant répertorie les ressources permettant de bénéficier du service clients, des services logiciels, de la formation et d'informations sur les produits et solutions.

Tableau 123. Ressources IBM

Ressource	Description et emplacement
IBM Support Portal	Vous pouvez personnaliser les informations de support en choisissant les produits et les rubriques qui vous intéressent à l'adresse www.ibm.com/support/entry/portal/Software/Information_Management/InfoSphere_Information_Server
Services logiciels	Vous trouverez des informations sur les services de conseil en logiciels, en technologies de l'information et en entreprise sur le site de solutions à l'adresse www.ibm.com/businesssolutions/
My IBM	Vous pouvez gérer les liens d'accès vers des sites IBM et des informations de support technique qui correspondent à vos besoins en créant un compte sur le site My IBM à l'adresse www.ibm.com/account/
Formation et certification	Pour en savoir plus sur les services de formation technique conçus pour les particuliers, les entreprises et les organisations publiques pour les aider dans l'achat, la gestion et l'optimisation de leurs compétences en matière de technologies de l'information, visitez le site http://www.ibm.com/software/sw-training/
Interlocuteurs IBM	Pour en savoir plus sur les solutions, vous pouvez contacter un interlocuteur IBM à l'adresse www.ibm.com/connect/ibm/us/en/

Support technique des produits de fédération, de réplication et de publication d'événements

Pour bénéficier du support technique, consultez les adresses suivantes :

- IBM InfoSphere Federation Server
www.ibm.com/software/data/integration/support/federation_server/
- IBM InfoSphere Replication Server
www.ibm.com/software/data/integration/support/replication_server/
- IBM InfoSphere Data Event Publisher
www.ibm.com/software/data/integration/support/data_event_publisher/

Support technique des produits Classic

Pour bénéficier du support technique, consultez les adresses suivantes :

- IBM InfoSphere Classic Federation Server for z/OS
www.ibm.com/software/data/integration/support/classic_federation_server_z/
- IBM InfoSphere Classic Replication Server for z/OS
www.ibm.com/software/data/infosphere/support/replication-server-z/
- IBM InfoSphere Classic Data Event Publisher for z/OS
www.ibm.com/software/data/integration/support/data_event_publisher_z/
- IBM InfoSphere Data Integration Classic Connector for z/OS
www.ibm.com/software/data/integration/support/data_integration_classic_connector_z/

Transmettre vos commentaires

Le tableau suivant décrit comment transmettre des commentaires à IBM sur les produits et la documentation.

Tableau 124. Transmettre vos commentaires à IBM

Type de commentaire	Action
Commentaires sur le produit	Vous pouvez transmettre des commentaires généraux sur les produits via l'étude Consumability Survey à l'adresse www.ibm.com/software/data/info/consumability-survey
Commentaires sur la documentation	Pour laisser un commentaire sur le centre de documentation, cliquez sur le lien Commentaires en haut à droite dans n'importe quelle rubrique du centre de documentation. Vous pouvez également envoyer des commentaires sur les manuels au format PDF, le centre de documentation ou d'autres documentations en suivant l'une des procédures suivantes : <ul style="list-style-type: none">• Formulaire de commentaire en ligne : www.ibm.com/software/data/rcf/• Courriel : comments@us.ibm.com

Lecture des diagrammes de syntaxe

Les règles suivantes s'appliquent aux diagrammes de syntaxe utilisés dans ce document :

- Lisez les diagrammes de syntaxe de gauche à droite, de haut en bas en suivant le chemin de la ligne. Les conventions suivantes sont utilisées :
 - Le symbole >>--- indique le début d'un diagramme de syntaxe.
 - Le symbole ---> indique que le diagramme de syntaxe continue sur la ligne suivante.
 - Le symbole >--- indique qu'un diagramme de syntaxe a été commencé sur la ligne précédente.
 - Le symbole --->< indique la fin du diagramme de syntaxe.
- Les éléments obligatoires apparaissent sur la ligne horizontale (chemin principal).



- Les éléments facultatifs apparaissent sous le chemin principal.



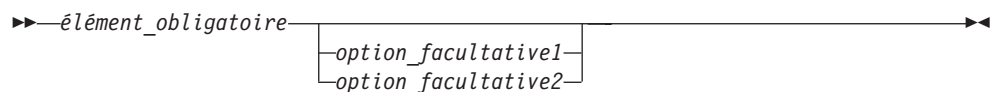
Si un élément facultatif apparaît au-dessus du chemin principal, il n'a pas d'incidence sur l'exécution de l'élément de syntaxe et est utilisé uniquement pour faciliter la lecture des données.



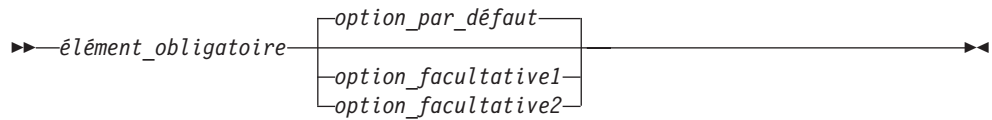
- Si vous pouvez faire une sélection parmi plusieurs éléments, ceux-ci apparaissent à la verticale, dans une pile.
Si vous devez sélectionner l'un des éléments, un seul élément de la pile apparaît dans le chemin principal.



Si la sélection de l'un des éléments est facultative, l'ensemble de la pile apparaît sous le chemin principal.



Si l'un des éléments est l'élément par défaut, il apparaît au-dessus du chemin principal et les options restantes sont affichées ci-dessous.



- Une flèche orientée à gauche, au-dessus de la ligne principale, indique un élément qui peut être répété.



Si la flèche de répétition contient une virgule, vous devez séparer les éléments répétés avec une virgule.

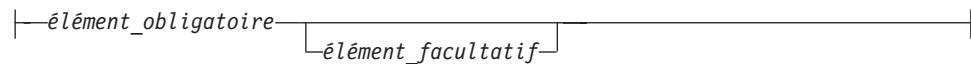


Une flèche de répétition au-dessus d'une pile indique que vous pouvez répéter les éléments de la pile.

- Un diagramme peut parfois être divisé en fragments. Le fragment de syntaxe est affiché séparément du diagramme de syntaxe principal mais le contenu du fragment doit être lu comme s'il se trouvait dans le chemin principal du diagramme.



Nom-Fragment :



- Les mots clés et les abréviations minimales associées apparaissent en majuscules. Vous devez les orthographier correctement.
- Les variables apparaissent en majuscules et en italique (par exemple, **column-name**). Elles représentent les noms ou les valeurs définis par l'utilisateur.
- Séparez les mots clés et les paramètres par au moins un espace si aucun signe de ponctuation n'apparaît dans le diagramme.
- Entrez les signes de ponctuation, les parenthèses, les opérateurs arithmétiques et les autres symboles tels qu'ils sont indiqués dans le diagramme.
- Les notes de pied de page sont indiquées par un numéro entre parenthèses, par exemple (1).

Remarques et marques

Remarques

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Pour le Canada, veuillez adresser votre courrier à :

IBM Director of Commercial Relations
IBM Canada Ltd.
3600 Steeles Avenue East
Markham, Ontario
L3R 9Z7
Canada

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues par écrit à l'adresse suivante :

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

Le paragraphe ci-dessous ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales : LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFAÇON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003 U.S.A.

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.

Ces informations sont fournies uniquement à titre de planification. Elles sont susceptibles d'être modifiées avant la mise à disposition des produits décrits.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

LICENCE DE COPYRIGHT :

Le présent logiciel contient des exemples de 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 lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Les exemples de programme sont fournis "EN L'ETAT", sans garantie aucune. IBM ne sera en aucun cas responsable des dommages liés à l'utilisation de ces exemples de programmes.

Toute copie totale ou partielle de ces exemples de programmes et des oeuvres qui en sont dérivées doit comprendre une notice de copyright, libellée comme suit :

© (nom de votre société) (année). Des segments de code sont dérivés des Programmes exemples d'IBM Corp. © Copyright IBM Corp. _entrez l'année ou les années_. All rights reserved.

Si vous visualisez ces informations en ligne, il se peut que les photographies et illustrations en couleur n'apparaissent pas à l'écran.

Marques

IBM, le logo IBM et ibm.com sont des marques d'International Business Machines Corp. dans de nombreux pays. Les autres noms de produits et de services peuvent appartenir à IBM ou à des tiers. La liste actualisée de toutes les marques d'IBM est disponible sur le Web à l'adresse www.ibm.com/legal/copytrade.shtml.

Les termes qui suivent sont des marques d'autres sociétés :

Adobe est une marque de Adobe Systems Incorporated aux Etats-Unis et/ou dans d'autres pays.

IT Infrastructure Library est une marque de The Central Computer and Telecommunications Agency qui fait désormais partie de The Office of Government Commerce.

Intel, le logo Intel, Intel Inside, le logo Intel Inside, Intel Centrino, le logo Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium, et Pentium sont des marques d'Intel Corporation ou de ses filiales aux Etats-Unis et dans certains autres pays.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

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

ITIL est une marque de The Office of Government Commerce et est enregistrée au bureau américain Patent and Trademark Office

UNIX est une marque enregistrée de The Open Group aux États-Unis et/ou dans certains autres pays.

Cell Broadband Engine est une marque de Sony Computer Entertainment, Inc. aux États-Unis et/ou dans certains autres pays. Elle est utilisée sous licence.

Java ainsi que tous les logos et toutes les marques incluant Java sont des marques de Sun Microsystems, Inc. aux États-Unis et/ou dans certains autres pays.

CASS, CASS Certified, DPV, LACS^{Link}, ZIP, ZIP + 4, ZIP Code, Post Office, Postal Service, USPS et United States Postal Service sont des marques d'United States Postal Service. IBM Corporation est un licencié non exclusif des marques DPV et LACS^{Link} d'United States Postal Service.

Les autres noms de sociétés, de produits et de services peuvent appartenir à des tiers.

Marques

Les marques IBM et certaines marques non IBM sont repérées lors de leur première apparition dans ce document avec le symbole adéquat.

IBM, le logo IBM et ibm.com sont des marques ou des marques d'International Business Machines Corp. dans de nombreux pays. Les autres noms de produits et de services peuvent appartenir à IBM ou à des tiers. La liste actualisée de toutes les marques d'IBM est disponible sur la page Web "Copyright and trademark information" à l'adresse www.ibm.com/legal/copytrade.shtml.

Les termes qui suivent sont des marques d'autres sociétés :

Adobe, le logo Adobe, PostScript, le logo PostScript sont des marques d'Adobe Systems Incorporated aux Etats-Unis et/ou dans certains autres pays.

IT Infrastructure Library est une marque de The Central Computer and Telecommunications Agency, qui fait désormais partie de The Office of Government Commerce.

Intel, le logo Intel, Intel Inside, le logo Intel Inside, Intel Centrino, le logo Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium, et Pentium sont des marques d'Intel Corporation ou de ses filiales aux Etats-Unis et dans certains autres pays.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

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

ITIL est une marque déposée et une marque déposée communautaire de l'Office of Government Commerce et est déposée auprès du Patent and Trademark Office américain.

UNIX est une marque enregistrée de The Open Group aux Etats-Unis et/ou dans certains autres pays.

Cell Broadband Engine est une marque de Sony Computer Entertainment, Inc. aux Etats-Unis et/ou dans certains autres pays. Elle est utilisée sous licence.

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.

Les autres noms de sociétés, de produits et de services peuvent appartenir à des tiers.

Index

A

abonnements
niveau schéma 125
arrêt 138
changement des options de niveau
table 137
création 132
création de profils 134
DATA CAPTURE CHANGES
et 135
suppression 140
pour une réplication DDL 125
abonnements de niveau schéma 125
arrêt 138
changement des options de niveau
table 137
création 132
création de profils 134
DATA CAPTURE CHANGES et 135
démarrage 135
options de niveau table 134
suppression 140
abonnements Q 270
ajout de colonnes à répliquer 141,
144, 295, 298
bidirectionnelle
création 106
changement des propriétés 293
Cibles CCD 212
clauses WHERE 73
colonnes de clé pour les cibles 87
colonnes de l'index pour les cibles 87
conditions de recherche 73
conditions inattendues dans les
cibles 89
création 69
réplication entre homologues,
présentation 114
réplication fédérée 158, 172
réplication unidirectionnelle,
présentation 66
dans une réplication bidirectionnelle
détection de conflit 110
démarrage 268
détection de conflit, réplication
bidirectionnelle 110
établissement de sous-ensembles de
colonnes source 72
filtrage de lignes 73
niveau schéma
démarrage 135
options
mappage des colonnes source et
cible 78
messages Q Capture 72
options concernant les erreurs 93
placer en mode déchargement 308
procédures stockées
description 181
paramètre, mapper vers des
colonnes source 188

abonnements Q (*suite*)
procédures stockées (*suite*)
paramètres, colonnes
supprimées 187
paramètres, identification de la
transaction 188
paramètres, type d'opération 185
présentation 181
profils objet cible 67
redémarrage sans redéfinition 307
règles d'attribution de nom 369
regroupement avec des mappes de
files de réplication 62
réinitialisation 293
réplication bidirectionnelle 97
réplication entre homologues 99
création 115
réplication unidirectionnelle 61
restrictions
colonnes IDENTITY 69, 106, 115,
116
restrictions sur le type de
données 233
suppression 309
activation
messages d'abonnement 596
ALTER COLUMN SET DATA TYPE -
opérations 146
ALTER PUB, commande 313
Analyseur de réplication Q
description 475
paramètres 475
anciennes valeurs de données
incluses dans les publications 231
applications utilisateur
liste des messages provenant de 593
apply_path, paramètre
description 418
ARM (Automatic Restart Manager) 362
arrêt
abonnements Q 270
publications 272
ASNCLP, commandes
création d'un fichier de
configuration 169
enregistrement des scripts SQL 243
asnpwd 441
ansnrt 445
asnsdrop 448
asntrc 465
attribut DATA CAPTURE CHANGES 37
aucun chargement
définition 191
description 199
autorisations
Centre de réplication et ASNCLP 5
Oracle LogMiner 168
pour la réplication, Linux, UNIX,
Windows 4
pour la réplication, présentation 4
pour la réplication, z/OS 4

autorisations (*suite*)
pour le programme Q Apply 4
pour le programme Q Capture 4
WebSphere MQ 32
moniteur d'alertes de
réplication 32
programme Q Apply 32
Q Capture, programme 32

B

bases de données cibles
configuration 48
bases de données DB2
réplication
versions antérieures 38, 56
bases de données DB2 distantes
réplication 38, 56
bases de données source
activation pour la réplication 47
configuration d'Oracle 165

C

CAPTURE_PATH, paramètre
spécification sous z/OS 355
captureupto, paramètre
Q Capture 407
Centre de réplication
enregistrement des scripts SQL 243
exécution ou enregistrement des
commandes 245
profils objet cible 67
changements de type de données,
réplication 146
chargement automatique
considérations z/OS 194
définition 191
définition de pseudonymes 194
présentation 192
utilitaires utilisés pour 192
chargement de tables non DB2 158
chargement manuel
définition 191
chargements parallèles
z/OS 194
cibles fédérées
configuration logicielle requise 151
création des objets requis 154
présentation 149
présentation de la tâche 151
restrictions 151
cibles Oracle, utilisation de
SQL*Plus 158, 555
cibles Sybase
liaison de packages Q Apply 50, 153
clause WHERE
filtrage des données
lignes dans des publications 227
lignes dans les abonnements Q 73

- clauses EDITPROC 233
- clauses FIELDPROC 233
- clauses VALIDPROC 233
- clés cible
 - abonnements Q 87
 - publications 230
 - réplication Q 87
- clients, service
 - contact 599
- clients WebSphere MQ 26
- colonne de gestion des versions 564
- colonne IS_KEY 87
- colonnes
 - colonnes de clé
 - abonnements Q 87
 - publication d'événements 230
 - publications 230
 - réplication Q 87
 - définition de sous-ensembles
 - abonnements Q 72
 - publication d'événements 226
 - publications 226
 - réplication Q 72
 - mappage
 - dans la réplication Q 78
 - dans les abonnements Q 78
 - publication
 - publication d'événements 226
 - publications 226
 - publiées dans les publications 230
 - réplication
 - abonnements Q 72
 - réplication Q 72
- colonnes calculées 80, 84
- colonnes d'identificateur de ligne
 - réplication 235
- colonnes de clé
 - pour la publication
 - d'événements 230
 - pour la réplication Q 87
 - pour les abonnements Q 87
 - pour les publications 230
- colonnes GENERATED ALWAYS 240
- colonnes identity 240
- colonnes source
 - ajout
 - publications existantes 315
 - réplication 144, 298
 - réplication unidirectionnelle 141, 295
 - mappage vers des colonnes cible
 - réplication Q 78
 - mappage vers une procédure
 - stockée 188
 - paramètre identifiant les colonnes
 - supprimées 187
 - publication d'un sous-ensemble
 - publication d'événements 226
 - publications 226
 - publiées dans une publication 230
 - réplication d'un sous-ensemble 72
- commande \$TA JES2 368
- commande ALTER QSUB 293
- commande asnoqcap 398
- commande asnoqcmd 412
- commande asnpwd
 - gestion de l'accès aux serveurs
 - distants 7
- commande asnqacmd 436
- commande asnqanalyze 475
- commande asnqapp
 - Q Apply 416
- commande asnqcap
 - Q Capture 374
- commande asnqccmd 407
- commande asnqmfmt 484
- commande asnqxfmt
 - réplication Q 486
- commande asnslst 449
- commande asntdiff 347, 450, 454, 462
 - exécution en mode parallèle 347
- commande asntrep 473
- Commande AT NetView 368
- Commande CREATE PUB 223
- Commande CREATE REPLQMAP 64
- commande CREER QSUB 69
- commande MODIFY 358
- commande REORG 337
- Commande START PUB 271
- Commande START QSUB 268
- commande startq 276, 288
- commande stop, Q Capture 264
- Commande stopq
 - Q Capture 278
- commande ulimit 45
- Commande VALIDATE WSMQ
 - ENVIRONMENT FOR 64, 220
- commandes
 - exécution et enregistrement (Centre de
 - réplication) 245
- commandes d'identification des incidents
 - asnqanalyze 475
- commandes de publication d'événements
 - asnoqcap 398
 - asnoqcmd 412
 - asnpwd 441
 - asnqanalyze 475
 - asnqcap 374
 - asnqccmd 407
 - asnqmfmt 484
 - asnqxfmt 486
 - asnsrct 445
 - asnsdrop 448
 - asnslist 449
 - asntdiff 347, 450, 454, 462
 - asntrc 465
 - asntrep 473
 - liste 373
 - organigramme 373
- commandes de réplication
 - asnoqcap 398
 - asnoqcmd 412
 - asnqacmd 436
 - asnqanalyze 475
 - asnqapp 416
 - asnqcap 374
 - asnqccmd 407
 - asnqmfmt 484
 - asnsrct 445
 - asnsdrop 448
 - asnslist 449
 - asntdiff 347, 450, 454, 462
 - asntrc 465
 - asntrep 473
- commandes de réplication Q
 - asnoqcap 398
 - asnoqcmd 412
 - asnqacmd 436
 - asnqanalyze 475
 - asnqapp 416
 - exemples d'utilisation 435
 - asnqcap 374
 - exemples d'utilisation 397
 - asnqccmd 407
 - asnqmfmt 484
 - asnsrct 445
 - asnsdrop 448
 - asnslist 449
 - asnsrct 445
 - asnsdrop 448
 - asnslist 449
 - asnsrct 445
 - asnsdrop 448
 - asnslist 449
 - asntdiff 347, 450, 454, 462
 - asntrep 473
 - liste 373
 - organigramme 373
- commandes de réplication SQL
 - asnpwd 441
 - asnsrct 445
 - asnsdrop 448
 - asntrc 465
- commandes système
 - asnoqcap 398
 - asnoqcmd 412
 - asnpwd 441
 - asnqacmd 436
 - asnqanalyze 475
 - asnqapp 416
 - asnqcap 374
 - asnqccmd 407
 - asnqmfmt 484
 - asnqxfmt 486
 - asnsrct 445
 - asnsdrop 448
 - asnslist 449
 - asntdiff 347, 450, 454, 462
 - asntrc 465
 - asntrep 473
- comportement d'accès CURRENTLY
 - COMMITTED 195
- CONCURRENT_ACCESS
 - _RESOLUTION 195
- conditions inattendues dans les cibles
 - abonnements Q 89
 - gestion avec des procédures
 - stockées 185
 - réplication unidirectionnelle 89
- configuration
 - ID utilisateur 3
 - mots de passe 3
 - réplication multidirectionnelle 97
 - réplication unidirectionnelle 61
- conflits
 - basés sur la valeur et basés sur la
 - version 327
 - dans la table
 - IBMQREP_EXCEPTIONS 327
- conflits basés sur la valeur 327
- conflits basés sur la version 327
- consignation complémentaire
 - activation 165
- consignation complémentaire au niveau
 - des tables 165

- consignation complémentaire
 - minimale 165
- consignation des archives
 - configuration 47
- contraintes référentielles
 - réplication entre homologues 99
- contrôle
 - tables de modification cohérente des données (CCD) 212
- conventions d'attribution de nom 369
 - pour la réplication Q et la publication d'événements 369
 - profils d'objet cible pour la réplication Q 67
- conversion
 - données pour la publication d'événements
 - description 2
 - données pour la réplication Q
 - description 1
- conversion de données
 - publication d'événements,
 - description 2
 - réplication Q, description 1
- CREATE PUBQMAP, commande 220
- création 220
 - abonnements Q
 - réplication entre homologues,
 - présentation 114
 - réplication unidirectionnelle,
 - présentation 66
 - objets WebSphere MQ de
 - réplication 9
 - publications
 - présentation 223
- critère d'alerte QUEUE_DEPTH 26

D

- DATA CAPTURE CHANGES et 135
- DB2 Extensions
 - restrictions 235
- DDL
 - modifications des tables source 129, 300
- définition de sous-ensembles
 - colonnes source
 - abonnements Q 72
 - publications 226
 - lignes source
 - abonnements Q 73
 - publications 227
- définition de sous-ensembles verticaux (colonne)
 - abonnements Q 72
 - publication d'événements 226
 - publications 226
 - réplication Q 72
- définitions de serveur
 - réplication fédérée 154
- démarrage
 - points connus dans les journaux 249
 - Q Apply, programme 416
- démarrage à chaud
 - effet sur les données de journal 332
 - paramètre startmode 376

- démarrage à chaud (*suite*)
 - paramètre startmode, trois types 374, 398
- démarrages à froid
 - paramètre startmode 376
 - prévention 254
- désactivation
 - message d'abonnement 596
- détection de conflit
 - réplication bidirectionnelle
 - options 110
 - présentation 97
 - réplication entre homologues 99
- dictionnaires de compression
 - z/OS 334
- données historisées
 - tables CCD 212
- données spatiales
 - restrictions 233

E

- élagage
 - Q Apply, programme
 - commande asnqacmd 436
- Élément col dans un message de schéma
 - d'abonnement 588
- élément racine msg
 - messages XML 569, 594
- encapsuleurs pour la réplication
 - fédérée 154
- erreurs
 - gestion de la logique pour les
 - procédures stockées 185
 - messages 586
 - récupération 338
 - réplication entre homologues 99
 - traitement dans la réplication unidirectionnelle 89
 - traitement destiné à la réplication
 - Q 93
- erreurs SQL
 - acceptables 327
 - imprévues 327
- établissement d'un sous-ensemble (horizontal) de lignes
 - abonnements Q 73
 - conditions de recherche 73
 - publication d'événements 227
 - réplication Q 73
- établissement d'un sous-ensemble (vertical) de colonnes
 - abonnements Q 72
 - publication d'événements 226
 - publications 226
 - réplication Q 72
- états SQL
 - acceptés dans la réplication Q
 - configuration pour une réplication entre homologues 99
- exceptions
 - réplication 327
- exceptions de données, généralités 327
- exemple de commandes
 - objets WebSphere MQ 23
- exemples MQSC 23
- EXPORT et LOAD, utilitaires 192

- expressions
 - réplication Q 80
 - SQL 80
 - XML 84
- expressions SQL 80, 84
- expressions SQL au niveau
 - des octets 80, 84

F

- fichier db2dj.ini 153
- fichier de mots de passe 7
 - création 441, 465
- fichiers journaux
 - recherche du plus ancien requise 333
- files d'attente
 - administration 23
 - pour Q Apply 23
 - pour Q Capture 23
 - déversement 23
 - envoi 23
 - réception
 - description 23
 - redémarrage 23
- files d'attente d'administration 23
 - paramètres requis 18
 - Q Apply 23
 - Q Capture 23
- files d'attente d'envoi
 - arrêt 278
 - démarrage 276
 - description 23
 - gestion des activités par file
 - d'attente 273
 - paramètres requis 18
 - points de redémarrage 273
 - règles d'attribution de nom 369
 - réinitialisation 304, 317
 - travailler avec Q Capture
 - dans la publication
 - d'événements 219
 - dans la réplication Q 62
- files d'attente de déversement
 - description 23
 - paramètres requis 18
- files d'attente de réception
 - arrêt 286
 - démarrage 288
 - description 23
 - paramètres requis 18
 - règles d'attribution de nom 369
 - réinitialisation 304
 - travailler avec Q Apply 62
- files d'attente de redémarrage
 - description 23
- filtrage
 - lignes
 - conditions de recherche,
 - publication d'événements 227
 - conditions de recherche, réplication
 - Q 73
- fonction HEX 80, 84
- Fonctionnalité LogMiner 165
- fonctionnalité Oracle LogMiner 165
 - configuration 166
 - configuration d'une base de données
 - source 168

fonctionnement
 programme Q Apply 436
 Q Apply, programme 281
 Q Capture, programme 247, 407, 412

G

GDPS Active/Active, configuration de la
 réplication Q pour 41
Gestionnaire de contrôle de service
 (SCM)
 affichage des services de
 réplication 366
 arrêt d'un service de réplication 365
 création de services de
 réplication 364
 démarrage d'un service de
 réplication 365
 description 363
 suppression de services de
 réplication 366
Gestionnaire de contrôle de service
 (SCM) Windows
 commande asnslist 449
 description 363
 liste des services de réplication 449
gestionnaire de files d'attente
 description 23
 paramètres requis 18
 regroupement 36

I

IBMQREP_APPLYENQ, table de
 contrôle 531
ID utilisateur
 stockage 7
identification et résolution des problèmes
 vérification du statut du
 programme 323
ignorer des transactions 259, 289
 Q Apply, programme 289
intégrité référentielle
 réplication entre homologues 99

J

JCL
 par lots, mode 358
 pour démarrer le moniteur d'alertes
 de réplication 357
 pour démarrer Q Apply 357
 pour démarrer Q Capture 356
journaux source
 conservation 332
 recherche manuelle du plus ancien
 requis (z/OS) 332

L

latence de bout en bout 326
lignes
 filtrage
 publication d'événements 227
 réplication Q 73

lignes en erreur 327
longueur de file d'attente pour valeurs
 d'objets LOB 35

M

maintenance 335
 environnement de publication
 d'événements
 objets source 331
 présentation 331
 tables de contrôle 335
 environnement de réplication Q
 démarrages à froid,
 prévention 254
 objets source 331
 présentation 331
 tables de contrôle 335
manipulation de données
 procédures stockées pour les
 abonnements Q
 description 181
 présentation 181
 réplication Q
 procédures stockées,
 description 181
 procédures stockées,
 présentation 181
mappage
 colonnes source et colonnes cible
 réplication Q 78
mappages utilisateur
 réplication fédérée 154
mappes de files de réplication
 création 64
 modification 304
 modification des noms de file
 d'attente 306
 regroupement avec des abonnements
 Q 62
 réplication bidirectionnelle 97
 réplication entre homologues 99
 réplication unidirectionnelle 61
 suppression 310
mappes de files pour la publication 220
 modification 317
 regroupement avec des
 publications 219
 suppression 318
marques 607
 liste de 603
mémoire
 WebSphere MQ 33
mémoires tampons de messages
 valeurs d'objet LOB 35
mentions légales 603
message d'abonnement désactivé 585
message d'ajout de colonne 592
message d'invalidation de la file d'attente
 d'envoi 595
message de chargement terminé 595
message de chargement terminé
 reçu 585
message de schéma d'abonnement 588
message XML d'opération de ligne 580
message XML de pulsation 587

messages
 colonnes incluses dans des
 publications 230
 d'un programme Q Capture
 lors de l'envoi d'un message par Q
 Capture pour les
 publications 226
 depuis un programme Q Capture
 pour les abonnements Q 72
 formatage et affichage 484, 486
 liste de messages provenant de Q
 Capture 568
 liste de types 567
 message d'abonnement désactivé 585
 message d'activation
 d'abonnement 596
 message d'ajout de colonne 592
 message d'invalidation de la file
 d'attente d'envoi 595
 message de chargement terminé 595
 message de chargement terminé
 reçu 585
 message de désactivation
 d'abonnement 596
 message de rapport d'erreurs 586
 message de schéma
 d'abonnement 588
 message de transaction 571
 message LOB 582
 messages d'information 567
 messages de contrôle 567
 messages de données 567
 publication d'événements
 vers Q Capture 593
 structure
 depuis Q Capture 568
 vers Q Capture 593
 valeurs avant dans les
 publications 231
 WebSphere MQ
 limite de taille 34
messages d'erreur
 rapports 586
messages d'information 567
messages de contrôle 567
messages de données 567
messages de redémarrage
 Q Capture 273
messages Q Capture
 colonnes incluses dans un
 message 230
 lors de l'envoi d'un message par Q
 Capture
 pour les abonnements Q 72
 pour les publications 226
 valeurs avant dans les
 publications 231
messages XML
 message d'opération de ligne 580
 message de pulsation 587
 msg : élément racine 569, 594
 présentation des types 567
 publication d'événements
 conditions techniques
 requis 567
 délimiteurs dans les données de
 type caractère 568

- messages XML (*suite*)
 - structure
 - présentation 567
- migration
 - bases de données
 - éviter 38, 56
- mode ARCHIVELOG
 - Oracle 165
- mode déchargement pour les abonnements Q 308
- modification
 - environnement de publication d'événements 313
 - environnement de réplication Q 293
 - paramètres de Q Capture 255
 - paramètres Q Apply 283
- moniteur d'alertes de réplication
 - commande MODIFY 358
 - démarrage avec JCL 357
 - exécution en mode de traitement par lots 358
 - packages
 - liaison 51
 - planification 366
 - Linux, UNIX 366
 - Windows 367
 - tâche démarrée 361
- mots de passe
 - configuration, présentation 3
 - stockage 7

O

- objets de canaux, paramètres requis 18
- OKSQLSTATES
 - réplication entre homologues 99
- opérations DDL, réplication 125, 132
- opérations de chargement
 - réplication de chargements de la table source 203, 205
- options de chargement
 - aucun 199
 - automatique 192
 - considérations z/OS 194
 - présentation 192
 - présentation 191
 - réplication bidirectionnelle 199
 - réplication entre homologues
 - deux serveurs 199
 - trois serveurs ou plus 199
 - réplication unidirectionnelle 199
 - sources Oracle 174
- options de chargement, réplication
 - tables consolidées 206
- Oracle
 - modifications de table source 177

P

- packages
 - liaison
 - moniteur d'alerte de réplication (Linux, UNIX, Windows) 51
 - présentation 49
 - programme Q Apply (Linux, UNIX, Windows) 50

- packages (*suite*)
 - liaison (*suite*)
 - programme Q Capture(Linux, UNIX, Windows) 50
- pages de codes 1
 - messages XML pour la publication d'événements 2
 - pour la publication d'événements
 - description 2
 - présentation 1
 - pour la réplication Q
 - description 1
 - présentation 1
 - programme Q Apply 1
 - Q Capture, programme
 - pour la publication d'événements 2
 - pour la réplication Q 1
- paires source-cible
 - mappage des colonnes, réplication Q 78
 - pour une réplication bidirectionnelle
 - description 97
 - pour une réplication entre homologues
 - description 99
 - pour une réplication unidirectionnelle 61
- par lots, mode
 - JCL 358
- paramètre, Q Capture 374
- paramètre add_partition
 - Q Capture (Linux, UNIX, Windows) 374, 376
- paramètre adminq
 - Q Capture 376
- paramètre adminq, Q Capture 374
- paramètre apply_path
 - Q Apply 416
- paramètre apply_schema
 - paramètres 418
 - travailler avec un programme Q
 - Apply en cours d'exécution 436
- paramètre apply_server
 - démarrer un programme Q
 - Apply 416
- paramètre applydelay
 - Q Apply 416, 418
- paramètre applyupto 416, 418
- paramètre applyupto, Q Apply 418
- paramètre arm
 - Q Apply 418
- paramètre autostop
 - démarrer un programme Q
 - Apply 416
 - démarrer un programme Q
 - Capture 398
 - description des paramètres Q
 - Capture 376
 - descriptions des paramètres Q
 - Apply 418
- paramètre autostop, Q Capture 374
- paramètre caf
 - Q Apply 416, 418
- paramètre caf, Q Capture 374
- paramètre capture_path 374

- paramètre capture_path, Q Capture 376, 398
- paramètre capture_schema 398
- paramètre capture_schema, Q
 - Capture 374, 376
- paramètre capture_server 374, 398
- paramètre capture_server, Q
 - Capture 376
- paramètre captureupto, Q Capture 264
- paramètre chgparms
 - Q Apply 436
 - Q Capture 407, 412
- paramètre CHLTYPE
 - WebSphere MQ 18
- paramètre classic_load_file_size
 - Q Apply 416
- paramètre classic_load_file_sz
 - Q Apply 416, 418
- paramètre commit_count
 - Q Apply 416, 418
- paramètre commit_interval
 - Q Capture 376
- paramètre commit_interval, Q
 - Capture 374, 398
- paramètre d'arrêt
 - Q Apply 436
 - Q Capture 407, 412
- paramètre d'élagage
 - Q Apply 436
 - Q Capture 407, 412
- paramètre d'état
 - Q Apply 436
 - Q Capture 407, 412
- paramètre data_applied, Q Capture 264
- paramètre data_sent, Q Capture 264
- paramètre de configuration du gestionnaire de base de données
 - maxagents 45
- paramètre de configuration logarchmeth1
 - paramètres 47
- paramètre de configuration maxappls
 - problèmes de mémoire 45
 - Q Apply 48
- paramètre de suppression
 - procédure stockée pour la réplication
 - Q 187
- paramètre deadlock_retries
 - Q Apply 416, 418
- paramètre DEFPSIST, WebSphere MQ 18
- paramètre DEFLOPT, WebSphere MQ 18
- paramètre DEFTYPE, WebSphere MQ 18
- paramètre dftmodelq
 - Q Apply 418
- paramètre DISCONT, WebSphere MQ 18
- paramètre HBINT
 - WebSphere MQ 18
- paramètre ignbaddata
 - Q Apply 418
- paramètre ignore_transid 107
- paramètre igntrig, Q Capture 374
- paramètre insert_bidi_signal
 - Q Apply 416, 418
- paramètre load_data_buff_size, Q
 - Apply 416
- paramètre load_data_buff_sz, Q
 - Apply 416, 418

- paramètre loadcopy_path
 - Q Apply 416, 418
- paramètre lob_send_option, Q
 - Capture 374
- paramètre logmarkertz
 - Q Apply 416, 418
- paramètre logreuse
 - Q Apply 416, 418
 - Q Capture 376
- paramètre logstdout
 - Q Apply 416, 418
 - Q Capture 376
- paramètre logstdout, Q Capture 374, 398
- paramètre lsn 249
- paramètre max_parallel_loads, Q
 - Apply 416, 418
- paramètre maxcmtseq 249
- paramètre MAXDEPTH
 - WebSphere MQ 18
- paramètre MAXMSGL
 - WebSphere MQ 18
- paramètre maxuproc 45
- paramètre MCAUSER 26
- paramètre memory_limit
 - Q Capture 376
- paramètre memory_limit, Q
 - Capture 374, 398
- paramètre message_persistence
 - Q Capture 376
- paramètre message_persistence, Q
 - Capture 374
- paramètre monitor_interval
 - Q Apply 416, 418
 - Q Capture 376
- paramètre monitor_interval, Q
 - Capture 374, 398
- paramètre monitor_limit
 - Q Apply 416, 418
 - Q Capture 374, 376
- paramètre monitor_limit, Q Capture 398
- paramètre MSGDLVSQ
 - WebSphere MQ 18
- paramètre nickname_commit_count
 - Q Apply 416, 418
- paramètre nickname_commit_ct
 - Q Apply 416, 418
- paramètre NUMTCB
 - chargements z/OS 194
- paramètre oracle_empty_str
 - Q Apply 418
- paramètre override_restartq, Q
 - Capture 252
- paramètre p2p_2nodes
 - Q Apply 416, 418
- paramètre part_hist_limit, Q
 - Capture 374
- paramètre prune_interval
 - Q Apply 416
 - Q Capture 374, 376, 398
- paramètre pwdfile
 - Q Apply 416, 418
- paramètre pwdfile, Q Capture 374, 398
- paramètre qmgr
 - Q Apply 416
 - Q Capture 376
- paramètre qryparms
 - Q Apply 436
- paramètre qryparms (*suite*)
 - Q Capture 407, 412
- paramètre reinit
 - Q Capture 407, 412
- paramètre reinitq
 - Q Apply 436
 - Q Capture 407, 412
- paramètre report_exception
 - Q Apply 418
- paramètre restartq
 - Q Capture 376
- paramètre restartq, Q Capture 374
- paramètre resumesub
 - Q Apply 436
- paramètre richklvl 416, 418
 - Q Apply 418
- paramètre SHARE
 - WebSphere MQ 18
- paramètre signal_limit
 - Q Capture 376
- paramètre signal_limit, Q Capture 374, 398
- paramètre skiptrans
 - Q Apply 416, 418, 436
- paramètre sleep_interval
 - Q Capture 376
- paramètre sleep_interval, Q
 - Capture 374, 398
- paramètre spill_commit_count
 - Q Apply 416, 418
- paramètre spillsub
 - Q Apply 436
- paramètre stale, Q Capture 374
- paramètre startallq 273
- paramètre startmode
 - Q Capture 376
- paramètre startmode, Q Capture 374, 398
- paramètre startq
 - Q Apply 436
- paramètre stopafter, Q Capture 264
- paramètre stopq
 - Q Apply 436
- paramètre term 418
 - Q Apply 416
 - Q Capture 376
- paramètre term, Q Capture 374, 398
- paramètre trace_ddl
 - Q Apply 418
- paramètre trace_limit
 - Q Apply 416, 418
 - Q Capture 376
- paramètre trace_limit, Q Capture 374, 398
- paramètre warntxsz
 - Q Capture 376
- paramètres
 - appel
 - programme Q Apply 416
 - programme Q Capture 374
 - Programme Q Capture 398
 - CAPTURE_PATH, spécification sous z/OS 355
 - procédures stockées, réplication
 - identification de la transaction 188
- paramètres (*suite*)
 - procédures stockées, réplication (*suite*)
 - identification des colonnes supprimées 187
 - identification du type d'opération 185
 - mappage vers des colonnes de clé, abonnements Q 87
 - mapper vers des colonnes source 188
 - réplication Q, description 181
 - réplication Q, présentation 181
 - Q Apply, programme
 - apply_path 418
 - apply_schema 418
 - apply_server 418
 - applydelay 416, 418
 - autostop 418
 - classic_load_file_sz 416, 418
 - commit_count 416, 418
 - deadlock_retries 418
 - load_data_buff_sz 416
 - load_data_buff_size 418
 - loadcopy_path 416, 418
 - logmarkertz 416, 418
 - logreuse 418
 - logstdout 418
 - max_parallel_loads 416, 418
 - modification 283
 - modification en cours d'exécution 436
 - monitor_interval 418
 - monitor_limit 418
 - nickname_commit_ct 416, 418
 - nmi_enable 418
 - nmi_socket_name 418
 - prune_batch_size 416
 - prune_method 416
 - pwdfile 418
 - qmgr 418
 - report_exception 418
 - skiptrans 416, 418
 - spill_commit_count 416, 418
 - term 418
 - trace_ddl 418
 - trace_limit 418
 - Q Capture, programme
 - add_partition 376
 - adminq 376
 - autostop 376
 - caf 376
 - capture_path 376
 - capture_schema 376
 - capture_server 376
 - commit_interval 376
 - descriptions 376
 - logreuse 376
 - logstdout 376
 - memory_limit 376
 - message_persistence 376
 - méthodes de changement 255
 - modification 255
 - modification en cours d'exécution 407
 - modification en cours de fonctionnement 412
 - monitor_interval 376

- paramètres (*suite*)
 - Q Capture, programme (*suite*)
 - monitor_limit 376
 - nmi_enable 376
 - nmi_socket_name 376
 - prune_interval 376
 - qfull_num_retries 376
 - qfull_retry_delay 376
 - qmgr 376
 - restartq 376
 - signal_limit 376
 - sleep_interval 376
 - startmode 376
 - term 376
 - trace_limit 376
 - warntsiz 376
 - WebSphere MQ 18
- paramètres d'opération
 - procédures stockées 185
- paramètres transactionnels, procédure stockée pour la réplication Q 188
- partitions de base de données multiples 208
- planification
 - exigences de stockage, WebSphere MQ 33
- Procédure stockée DSNUTILS pour z/OS 194
- procédures stockées
 - abonnements Q
 - conditions inattendues dans les cibles 89
 - description 181
 - options concernant les erreurs 93
 - paramètres pour les colonnes de clé 87
 - présentation 181
 - écriture 183
 - paramètres
 - identification de la transaction 188
 - identification des colonnes supprimées 187
 - identification du type d'opération 185
 - mapper vers des colonnes source 188
 - réplication Q
 - conditions inattendues dans les cibles 89
 - description 181
 - options concernant les erreurs 93
 - paramètres pour les colonnes de clé 87
 - présentation 181
- profils
 - profils d'objet cible pour la réplication Q 67
- profils objet cible 67
- programme du moniteur d'alertes de réplication.
 - vérification du statut 321
- programme Q Apply
 - arrêt, entre homologues (deux serveurs) 122
 - commande MODIFY 358
 - configuration fédérée 153
 - programme Q Apply (*suite*)
 - conversion de données 1
 - démarrage, entre homologues (deux serveurs) 119
 - démarrage, entre homologues (trois serveurs) 120, 122
 - démarrage avec JCL 357
 - exécution en mode de traitement par lots 358
 - mémoire sous AIX 45
 - mode déchargement pour les abonnements Q 308
 - pages de codes 1
 - planification 366
 - Linux, UNIX 366
 - Windows 367
 - procédures stockées
 - description 181
 - réplication bidirectionnelle 97
 - réplication entre homologues 99
 - schéma
 - règles de réplication bidirectionnelle 97
 - règles s'appliquant à la réplication entre homologues 99
 - tâche démarrée 361
 - travailler avec des files d'attente de réception 62
 - unités d'exécution agent
 - définies 323
 - unités d'exécution du navigateur
 - définies 323
 - vérification du statut 321
- programme Q Apply`
 - planification 368
 - z/OS 368
- programme Q Capture
 - arrêt 407
 - démarrage
 - commande asnqcap 374
- Programme Q Capture
 - paramètres
 - captureupto 264
- programmes de réplication
 - planification 366
- prune_batch_size, paramètre Q Apply 416
- prune_method, paramètre Q Apply 416
- pseudonymes
 - chargement automatique 194
- publication
 - avec le programme Q Capture
 - colonnes incluses dans une publication 230
 - lors de l'envoi d'un message par Q Capture 226
 - configuration de la publication d'événements 219
 - données converties 2
 - publication d'événements
 - colonnes incluses dans un message 230
 - lors de l'envoi d'un message par Q Capture 226
- publication (*suite*)
 - publications
 - colonnes incluses dans un message 230
 - lors de l'envoi d'un message par Q Capture 226
 - publications XML
 - valeurs avant 231
 - regroupement de publications et de mappes de files d'attente pour la publication 219
 - sous-ensemble de données
 - filtrage de lignes dans des publications 227
 - sous-ensembles de colonnes source pour les publications 226
 - valeurs avant dans les publications 231
- publication d'événements
 - ajout de colonnes à répliquer 315
 - autorisations nécessaires
 - Centre de réplication et ASNCLP 5
 - présentation 4
 - Q Capture 4
 - colonnes de clé pour les publications 230
 - colonnes incluses dans un message 230
 - colonnes source pour les publications 226
 - commande REORG 337
 - configuration 219
 - configuration des mots de passe 3
 - conversion de données
 - description 2
 - création de publications 223
 - dictionnaires de compression 334
 - erreurs, récupération 338
 - établissement de sous-ensembles de colonnes source 226
 - exigences de connectivité 6
 - exigences de stockage, WebSphere MQ 33
 - gestion
 - démarrages à froid, prévention 254
 - journaux source, conservation 332
 - journaux source, recherche manuelle du plus ancien requis (z/OS) 332
 - objets source 331
 - présentation 331
 - systèmes source 331
 - ignorer des transactions 259
 - lors de l'envoi d'un message par Q Capture 226
 - messages
 - liste de messages provenant de Q Capture 568
 - liste de types 567
 - message d'abonnement désactivé 585
 - message d'activation d'abonnement 596
 - message d'ajout de colonne 592

- publication d'événements (*suite*)
 - messages (*suite*)
 - message d'invalidation de la file d'attente d'envoi 595
 - message de chargement terminé 595
 - message de chargement terminé reçu 585
 - message de désactivation d'abonnement 596
 - message de rapport d'erreurs 586
 - message de schéma d'abonnement (subSchema) 588
 - message de transaction 571
 - message LOB 582
 - messages d'information 567
 - messages de contrôle 567
 - messages de données 567
 - structure, depuis Q Capture 568
 - structure, vers Q Capture 593
 - vers Q Capture 593
 - messages XML
 - conditions techniques
 - requis 567
 - délimiteurs dans les données de type caractère 568
 - message d'opération de ligne 580
 - message de pulsation 587
 - msg : élément racine 569, 594
 - présentation des types 567
 - structure, présentation 567
 - modification 313
 - nom d'objets
 - casse 372
 - règles d'attribution de nom 369
 - Objets WebSphere MQ requis
 - à distance 13
 - packages, liaison
 - moniteur d'alertes de réplication 51
 - présentation 49
 - Q Capture 50
 - pages de codes 2
 - plusieurs partitions de base de données 208
 - publications
 - clauses WHERE 227
 - conditions de recherche 227
 - filtrage de lignes 227
 - redéfinition de modules et de plans 340
 - regroupement de gestionnaires de files d'attente 36
 - regroupement de publications et de mappes de files d'attente pour la publication 219
 - restrictions, types de données 233
 - services système 355
 - source Oracle
 - mappage de colonne 178
 - types de données 178
 - stockage des ID utilisateur et des mots de passe 7
 - tables de contrôle
 - élagage 335
 - gestion 335
 - présentation 489

- publication d'événements (*suite*)
 - utilitaire RUNSTATS 337
 - valeurs avant incluses 231
 - vérification du statut 321
 - WebSphere MQ
 - configuration, présentation 9
 - objets 9
 - publications 272
 - ajout de colonnes à répliquer 315
 - changement des propriétés 313
 - clauses WHERE 227
 - colonnes de clé 230
 - colonnes incluses dans un message 230
 - conditions de recherche 227
 - création 223
 - présentation 223
 - démarrage 271
 - établissement de sous-ensembles de colonnes source 226
 - filtrage de lignes 227
 - messages Q Capture 226
 - règles d'attribution de nom 369
 - regroupement avec des mappes de files d'attente pour la publication 219
 - réinitialisation 313
 - suppression 316
 - valeurs avant incluses 231

Q

- Q Apply, programme
 - arrêt 285, 436
 - arrêt du traitement des messages 286, 436
 - autorisations nécessaires 4
 - commande asnqacmd 436
 - commande d'élagage 436
 - démarrage 281
 - commande asnqapp 416
 - démarrage du traitement des messages 288, 436
 - détection de conflit, réplication bidirectionnelle 110
 - fonctionnement 281
 - ignorer des transactions 289
 - options concernant les erreurs 93
 - packages, liaison 50
 - paramètres
 - apply_path 418
 - apply_schema 418
 - apply_server 418
 - applydelay 416, 418
 - applyupto 416, 418
 - arm 418
 - autostop 418
 - caf 416, 418
 - changement sauvegardé 284
 - classic_load_file_sz 416, 418
 - commit_count 416, 418
 - deadlock_retries 418
 - description 418
 - dftmodelq 416, 418
 - ignbaddata 416, 418
 - insert_bidi_signal 416, 418
 - load_data_buff_size 418

- Q Apply, programme (*suite*)
 - paramètres (*suite*)
 - load_data_buff_sz 416
 - loadcopy_path 416, 418
 - logmarkertz 416, 418
 - logreuse 418
 - logstdout 418
 - max_parallel_loads 416, 418
 - modification 283
 - modification dynamique 283
 - monitor_interval 418
 - monitor_limit 418
 - nickname_commit_ct 416, 418
 - nmi_enable 418
 - nmi_socket_name 418
 - oracle_empty_str 418
 - p2p_2nodes 416, 418
 - prune_batch_size 416
 - prune_interval 416, 418
 - prune_method 416
 - pwdfile 418
 - qmgr 418
 - report_exception 418
 - richklvl 416, 418
 - skiptrans 416, 418
 - spill_commit_count 416, 418
 - term 418
 - trace_ddl 418
 - trace_limit 418
 - paramètres d'appel 416
 - procédures stockées
 - présentation 181
 - réplication unidirectionnelle 61
 - traitement de conditions
 - inattendues 89
 - vérification des valeurs de paramètres d'exécution 436
 - vérification du statut 436
 - Q Apply, serveur
 - liste des tables de contrôle 530
 - Q Apply, tables de contrôle
 - liste 530
 - Q Capture
 - paramètres
 - startallq 273
 - Q Capture, arrêt à un point spécifié 264
 - Q Capture, programme 249
 - arrêt 263, 412
 - arrêt, entre homologues (deux serveurs) 122
 - autorisations nécessaires 4
 - colonnes envoyées dans des publications 230
 - commande d'élagage 407, 412
 - commande MODIFY 358
 - configuration pour les sources Oracle 165
 - conversion de données
 - pour la publication d'événements 2
 - pour la réplication Q 1
 - démarrage 247
 - commande asnoqcap 398
 - démarrage, entre homologues (deux serveurs) 119
 - démarrage, entre homologues (trois serveurs) 120, 122

- Q Capture, programme (*suite*)
 - démarrage avec JCL 356
 - démarrage pour les sources Oracle 176
 - démarrages à froid
 - paramètre startmode 376
 - prévention 254
 - envoi de commandes en cours d'exécution 407, 412
 - exécution en mode de traitement par lots 358
 - fonctionnement 247
 - gestion des modifications de la structure de table 129, 300
 - ignorer des transactions 259
 - lors de l'envoi d'un message par Q Capture
 - pour la publication 226
 - pour les abonnements Q 72
 - mémoire sous AIX 45
 - messages 593
 - messages XML 568
 - packages, liaison 50
 - pages de codes 1
 - pour la publication d'événements 2
 - paramètres
 - add_partition 376
 - adminq 376
 - autostop 376
 - capture_path 376
 - capture_schema 376
 - capture_server 376
 - changement sauvegardé 258
 - commit_interval 376
 - data_applied 264
 - data_sent 264
 - description 376
 - hs 376
 - igncasdel 376
 - igntrig 376
 - logread_prefetch 376
 - logreuse 376
 - logstout 376
 - memory_limit 376
 - message_persistence 376
 - méthodes de changement 255
 - modification 255
 - modification dynamique 257
 - monitor_interval 376
 - nmi_enable 376
 - nmi_socket_name 376
 - override_restartq 252, 376
 - part_hist_limit 376
 - prune_interval 376
 - qmgr 376
 - restartq 376
 - signal_limit 376
 - sleep_interval 376
 - stale 376
 - startallq 376
 - startmode 376
 - stopafter 264
 - term 376
 - trace_limit 376
 - trans_batch_sz 376
 - warnifi 376

- Q Capture, programme (*suite*)
 - paramètres (*suite*)
 - warnlogapi 376
 - warntsiz 376
 - planification 366, 368
 - Linux, UNIX 366
 - messages Q Capture 72
 - Windows 367
 - recherche du plus ancien journal requis 333
 - réinitialisation 293, 313, 407, 412
 - Réinitialisation d'une file d'attente d'envoi 407, 412
 - réplication bidirectionnelle 97
 - réplication entre homologues 99
 - réplication unidirectionnelle 61
 - schéma
 - règles de réplication bidirectionnelle 97
 - règles s'appliquant à la réplication entre homologues 99
 - tâche démarrée 361
 - travailler avec des files d'attente d'envoi
 - dans la publication d'événements 219
 - dans la réplication Q 62
 - travailler avec Oracle LogMiner 166
 - valeurs avant dans les publications 231
 - vérification des valeurs de paramètres d'exécution 407, 412
 - vérification du statut 321, 407, 412
 - z/OS 368
- Q Capture, schémas
 - règles d'attribution de nom 369
 - règles de réplication bidirectionnelle 97
 - règles s'appliquant à la réplication entre homologues 99
 - suppression 311, 319
- Q Capture, serveur
 - IBMQREP_EXCLSHEMA, table de contrôle 505
 - liste des tables de contrôle 489
- Q Capture, tables de contrôle
 - liste 489
- qualificatif de moniteur
 - règles d'attribution de nom 369

R

- recapture
 - éviter 107
- redémarrage
 - programme Q Capture 249
- regroupement d'objets de réplication
 - mappe de files de réplication et abonnements Q 62
 - regroupement de publications et de mappes de files d'attente pour la publication 219
- répertoires temporaires
 - variable d'environnement TMPDIR 44
- REPL_ADDCOL, option 141, 295

- réplication
 - abonnements Q
 - création destinée à la réplication entre homologues 114
 - création pour la réplication unidirectionnelle, présentation 66
 - messages de Q Capture 72
 - procédures stockées, description 181
 - procédures stockées, présentation 181
 - colonnes ROWID 235
 - données converties 1
 - mappage des colonnes source et cible
 - réplication Q 78
 - procédure stockée
 - d'administration 26
 - réplication bidirectionnelle
 - description 97
 - détection de conflit 110
 - réplication entre homologues 99
 - réplication multidirectionnelle, configuration 97
 - réplication unidirectionnelle
 - conditions inattendues 89
 - configuration 61
 - description 61
 - utilisation des procédures stockées, présentation 181
 - sous-ensemble de données
 - colonnes source destinées à la réplication Q 72
 - filtrage de lignes dans les abonnements Q 73
 - tables source
 - chargements 203, 205
 - type de données XML 237
 - types de données
 - XML 237
 - types de données d'objet BLOB 235
 - types de données d'objet CLOB 235
 - types de données d'objet DBCLOB 235
 - vers une procédure stockée dans un abonnement Q 188
- réplication bidirectionnelle
 - abonnements Q 106
 - ajout de colonnes à répliquer 144, 298
 - amélioration des performances 107
 - description 97
 - détection de conflit 97, 110
 - objets de réplication 97
 - objets WebSphere MQ 14
 - options concernant les erreurs 93
 - options de chargement 199
- réplication des changements de type de données 146
- réplication des opérations DDL 125, 132
- réplication entre homologues 115
 - abonnements Q
 - présentation 114
 - ajout de colonnes à répliquer 144, 298
 - arrêt 122
 - colonne de gestion des versions 564

- réplication entre homologues *(suite)*
 - démarrage 119, 120, 122
 - description 99
 - détection de conflit 99
 - deux serveurs
 - objets 99
 - options de chargement 199
 - deux serveurs distants
 - objets WebSphere MQ 14
 - intégrité référentielle 99
 - objets de réplication
 - deux serveurs 99
 - trois serveurs ou plus 99
 - options concernant les erreurs 93
 - trois serveurs distants ou plus
 - objets WebSphere MQ 16
 - trois serveurs ou plus
 - objets 99
 - options de chargement 199
- réplication multidirectionnelle
 - choix entre la réplication
 - bidirectionnelle et la réplication entre homologues 105
 - configuration 97
 - réplication bidirectionnelle 97
 - réplication entre homologues 99
- réplication Q
 - abonnements de niveau schéma 125, 132, 134, 135, 138
 - abonnements Q
 - clauses WHERE 73
 - conditions de recherche 73
 - filtrage de lignes 73
 - options concernant les erreurs 93
 - autorisations nécessaires
 - Centre de réplication et ASNCLP 5
 - pour Q Apply 4
 - pour Q Capture 4
 - présentation 4
 - chargement de tables consolidées 206
 - cibles fédérées
 - configuration logicielle requise 151
 - présentation 149
 - présentation de la tâche 151
 - restrictions 151
 - colonnes calculées 80, 84
 - commande REORG 337
 - configuration des mots de passe 3
 - conversion de données
 - description 1
 - détection de conflit
 - réplication bidirectionnelle 97
 - réplication entre homologues 99
 - détection de conflit, réplication
 - bidirectionnelle 110
 - erreurs, récupération 338
 - exigences de connectivité 6
 - exigences de stockage, WebSphere MQ 33
 - expressions 80
 - gestion
 - démarrages à froid, prévention 254
 - journaux source, conservation 332

- réplication Q *(suite)*
 - gestion *(suite)*
 - journaux source, recherche manuelle du plus ancien requis (z/OS) 332
 - objets source 331
 - présentation 331
 - systèmes source 331
 - ignorer des transactions 259, 289
 - modification 293
 - nom d'objets
 - casse 372
 - règles d'attribution de nom 369
 - options concernant les erreurs 93
 - packages
 - moniteur d'alerte de réplication (Linux, UNIX, Windows) 51
 - packages, liaison
 - présentation 49
 - Q Apply (Linux, UNIX, Windows) 50
 - Q Capture (Linux, UNIX, Windows) 50
 - pages de codes 1
 - plusieurs partitions de base de données 208
 - procédures stockées
 - paramètre, mapper vers des colonnes source 188
 - paramètres, colonnes supprimées 187
 - paramètres, identification de la transaction 188
 - paramètres, type d'opération 185
 - présentation 181
 - profils objet cible 67
 - Q Apply, programme
 - réplication bidirectionnelle 97
 - réplication entre homologues 99
 - réplication unidirectionnelle 61
 - Q Capture, programme
 - réplication bidirectionnelle 97
 - réplication entre homologues 99
 - réplication unidirectionnelle 61
 - recommandations relatives au chargement des tables cible 191
 - redéfinition de modules et de plans 340
 - regroupement de gestionnaires de files d'attente 36
 - regroupement de mappes de files de réplication et d'abonnements Q 62
 - réplication bidirectionnelle
 - description 97
 - détection de conflit 97, 110
 - réplication DDL 125
 - réplication entre homologues
 - colonne de gestion des versions 564
 - création d'abonnements Q, présentation 114
 - description 99
 - détection de conflit 99
 - réplication multidirectionnelle
 - choix entre la réplication
 - bidirectionnelle et la réplication entre homologues 105

- réplication Q *(suite)*
 - réplication multidirectionnelle *(suite)*
 - configuration 97
 - réplication bidirectionnelle 97
 - réplication entre homologues 99
 - réplication unidirectionnelle
 - colonnes de clé pour les cibles 87
 - colonnes de l'index pour les cibles 87
 - colonnes source destinées aux abonnements Q 72
 - conditions de recherche 73
 - conditions inattendues dans les cibles 89
 - configuration 61
 - création d'abonnements Q, présentation 66
 - description 61
 - établissement de sous-ensembles de colonnes source 72
 - filtrage de lignes 73
 - mappage des colonnes source et cible 78
 - messages Q Capture 72
 - restrictions
 - colonnes IDENTITY 69, 106, 115, 116
 - types de données 233
 - services système 355
 - source Oracle
 - mappage de colonne 178
 - types de données 178
 - sources Oracle 161
 - présentation de la tâche 162
 - restrictions 162
 - support client Oracle 164
 - stockage des ID utilisateur et des mots de passe 7
 - tables de contrôle
 - élagage 335
 - gestion 335
 - liste, Q Apply 530
 - liste, Q Capture 489
 - présentation 489
 - utilitaire RUNSTATS 337
 - vérification du statut 321
 - WebSphere MQ
 - configuration, présentation 9
 - objets 9
- Réplication Q
 - abonnements Q
 - règles d'attribution de nom 369
 - dictionnaires de compression 334
 - réplication unidirectionnelle
 - abonnements Q
 - clauses WHERE 73
 - colonnes de clé pour les cibles 87
 - colonnes de l'index pour les cibles 87
 - colonnes source 72
 - conditions de recherche 73
 - conditions inattendues dans les cibles 89
 - création, présentation 66
 - filtrage de lignes 73
 - mappage des colonnes source et cible 78

- réplication unidirectionnelle (*suite*)
 - abonnements Q (*suite*)
 - messages Q Capture 72
 - ajout de colonnes à répliquer 141, 295
 - configuration
 - description 61
 - regroupement de mappes de files de répllication et d'abonnements Q 62
 - description 61
 - Objets WebSphere MQ requis
 - à distance 9
 - même système 11
 - options concernant les erreurs 93
 - options de chargement 199
 - procédures stockées
 - description 181
 - présentation 181
- reprise des abonnements Q en mode déchargement 308
- restrictions
 - clauses EDITPROC 233
 - clauses FIELDPROC 233
 - clauses VALIDPROC 233
 - objets LOB de DB2 Extensions 235
 - type de données spatiales 233
 - types de données 233
 - types de données abstraits 233
 - types de données définis par l'utilisateur 233
 - types de données distincts 233
 - types de données LONG VARCHAR 233

S

- schéma Q Apply
 - règles d'attribution de nom 369
 - règles de répllication
 - bidirectionnelle 97
 - règles s'appliquant à la répllication entre homologues 99
 - suppression 311
- schémas
 - pour le programme Q Apply
 - règles de répllication
 - bidirectionnelle 97
 - règles s'appliquant à la répllication entre homologues 99
 - pour le programme Q Capture
 - règles de répllication
 - bidirectionnelle 97
 - règles s'appliquant à la répllication entre homologues 99
 - règles d'attribution de nom
 - pour une répllication
 - bidirectionnelle 97
 - pour une répllication entre homologues 99
- SCM (Service Control Manager)
 - affichage des services de répllication 366
 - arrêt de services de répllication 365
 - création de services de répllication 364

- SCM (Service Control Manager) (*suite*)
 - démarrage de services de répllication 365
 - description 363
 - suppression de services de répllication 366
- scripts ASNCLP
 - génératiion 243
- scripts SQL
 - enregistrement avec ASNCLP 243
 - enregistrement avec le Centre de répllication 243
- sérialisation
 - unités d'exécution 323
- Serveur de contrôle
 - IBMQREP_CAPQMON 500
- serveur Q Apply
 - Table de contrôle
 - IBMQREP_TARGETS 555
- Serveur Q Apply
 - configuration fédérée 153
 - IBMQREP_APPLYENQ, table de contrôle 531
 - Table de contrôle
 - IBMQREP_APPENVINFO 532
 - Table de contrôle
 - IBMQREP_APPLYMON 533
 - Table de contrôle
 - IBMQREP_APPLYPARMS 538
 - Table de contrôle
 - IBMQREP_APPLYTRACE 545
 - Table de contrôle
 - IBMQREP_DELTOMB 546
 - Table de contrôle
 - IBMQREP_DONEMSG 547
 - Table de contrôle
 - IBMQREP_EXCEPTIONS 547
 - Table de contrôle
 - IBMQREP_RECVQUEUEUES 550
 - table de contrôle
 - IBMQREP_SAVERI 552
 - Table de contrôle
 - IBMQREP_SPILLEDROW 554
 - Table de contrôle
 - IBMQREP_SPILLQS 554
 - Table de contrôle
 - IBMQREP_TRG_COLS 563
- serveur Q Capture
 - pages de codes pour la publication d'événements 2
 - Table de contrôle,
 - IBMQREP_COLVERSION 503
 - Table de contrôle
 - IBMQREP_ADMINMSG 490
 - Table de contrôle
 - IBMQREP_CAPENQ 491
 - Table de contrôle
 - IBMQREP_CAPENVINFO 492
 - Table de contrôle
 - IBMQREP_CAPMON 493
 - Table de contrôle
 - IBMQREP_CAPPARMS 495
 - Table de contrôle
 - IBMQREP_CAPQMON 500
 - Table de contrôle
 - IBMQREP_CAPTRACE 502

- serveur Q Capture (*suite*)
 - table de contrôle
 - IBMQREP_IGNTRAN 506
 - table de contrôle
 - IBMQREP_IGNTRANTRC 507
 - table de contrôle
 - IBMQREP_PART_HIST 507
 - table de contrôle
 - IBMQREP_SCHEMASUBS 509
 - Table de contrôle
 - IBMQREP_SENDQUEUEUES 511
 - Table de contrôle
 - IBMQREP_SIGNAL 515
 - Table de contrôle
 - IBMQREP_SRC_COLS 520
 - Table de contrôle
 - IBMQREP_SRCH_COND 521
 - Table de contrôle
 - IBMQREP_SUBS 522
 - table de contrôle
 - IBMQREP_SUBS_PROF 528
 - Table de contrôle
 - IBMQREP_TABVERSION 529
- serveur Q Capture pour Oracle
 - table de contrôle
 - IBMQREP_EOLFLUSH 504
- serveurs cible
 - profils objet cible 67
- serveurs source
 - lors de l'envoi d'un message par Q Capture
 - pour les abonnements Q 72
 - pour les publications 226
 - pages de codes
 - pour la publication d'événements 2
 - pour la répllication Q 1
- service
 - clients 599
- service Windows
 - création 445, 448
- services de répllication
 - affichage 366
 - arrêt 365
 - création 364
 - démarrage 365
 - description 363
 - liste 449
 - nom 363
 - nom d'affichage 363
 - suppression 366
- Services logiciels
 - contact 599
- SET DATA TYPE - opérations 146
- signal ADDCOL 141, 295
- signaux
 - ADDCOL 141, 144, 295, 298, 315
- sites Web
 - non IBM 601
- sources Oracle
 - chargement de tables cible 174
 - création d'un fichier de configuration
 - ASNCLP 169
 - présentation de la tâche 162
 - restrictions 162
- SQL*Plus 158, 555

SQLSTATE
 acceptés dans la réplication Q
 configuration pour une réplication
 entre homologues 99
 statut des programmes de réplication,
 vérification 321
 stopafter, paramètre
 Q Capture 407
 stopq, commande
 Q Apply 286
 support client Oracle 164
 synchronisation
 utilitaires asntdiff et asntrepare 341
 systèmes source
 définition 331
 gestion 331

T

Table de contrôle,
 IBMQREP_COLVERSION 503
 Table de contrôle
 IBMQREP_ADMINMSG 490
 Table de contrôle
 IBMQREP_APPENVINFO 532
 Table de contrôle
 IBMQREP_APPLYMON 533
 Table de contrôle
 IBMQREP_APPLYPARMS
 description 538
 Table de contrôle
 IBMQREP_APPLYTRACE 545
 Table de contrôle
 IBMQREP_CAPENQ 491
 Table de contrôle
 IBMQREP_CAPENVINFO 492
 Table de contrôle
 IBMQREP_CAPMON 493
 Table de contrôle IBMQREP_CAPPARMS
 description 495
 Table de contrôle
 IBMQREP_CAPTRACE 502
 Table de contrôle
 IBMQREP_DELTOMB 546
 Table de contrôle
 IBMQREP_DONEMSG 547
 table de contrôle
 IBMQREP_EOLFLUSH 504
 Table de contrôle
 IBMQREP_EXCEPTIONS 547
 table de contrôle
 IBMQREP_EXCLSHEMA 505
 table de contrôle
 IBMQREP_IGNTRAN 259, 506
 table de contrôle
 IBMQREP_IGNTRANTRC 259, 507
 table de contrôle
 IBMQREP_PART_HIST 507
 Table de contrôle
 IBMQREP_RECVQUEUEES 550
 Table de contrôle
 IBMQREP_SAVERI 552
 table de contrôle
 IBMQREP_SCHEMASUBS 509
 Table de contrôle
 IBMQREP_SENDQUEUEES 511

Table de contrôle
 IBMQREP_SIGNAL 515
 Table de contrôle
 IBMQREP_SPILLEDROW 554
 Table de contrôle
 IBMQREP_SPILLQS 554
 Table de contrôle
 IBMQREP_SRC_COLS 520
 Table de contrôle
 IBMQREP_SRCH_COND 521
 Table de contrôle IBMQREP_SUBS 522
 table de contrôle
 IBMQREP_SUBS_PROF 528
 Table de contrôle
 IBMQREP_TABVERSION 529
 Table de contrôle
 IBMQREP_TARGETS 555
 Table de contrôle
 IBMQREP_TRG_COLS 563
 table des différences 341
 Table IBMQREP_IGNTRAN 107
 table SYSIBM.SYSCOLUMNS 37
 table SYSIBM.SYSTABLEPART 37
 Table SYSIBM.SYSTABLES 37
 tables
 IBMQREP_ADMINMSG 490
 IBMQREP_APPENVINFO 532
 IBMQREP_APPLYENQ 531
 IBMQREP_APPLYMON 533
 IBMQREP_APPLYPARMS 538
 IBMQREP_APPLYTRACE 545
 IBMQREP_CAPENQ 491
 IBMQREP_CAPENVINFO 492
 IBMQREP_CAPMON 493
 IBMQREP_CAPPARMS 495
 IBMQREP_CAPQMON 500
 IBMQREP_CAPTRACE 502
 IBMQREP_COLVERSION 503
 IBMQREP_DELTOMB 546
 IBMQREP_DONEMSG 547
 IBMQREP_EOLFLUSH 504
 IBMQREP_EXCEPTIONS 547
 IBMQREP_EXCLSHEMA 505
 IBMQREP_IGNTRAN 506
 IBMQREP_IGNTRANTRC 507
 IBMQREP_PART_HIST 507
 IBMQREP_RECVQUEUEES 550
 IBMQREP_SAVERI 552
 IBMQREP_SCHEMASUBS 509
 IBMQREP_SENDQUEUEES 511
 IBMQREP_SIGNAL 515
 IBMQREP_SPILLEDROW 554
 IBMQREP_SPILLQS 554
 IBMQREP_SRC_COLS 520
 IBMQREP_SRCH_COND 521
 IBMQREP_SUBS 522
 IBMQREP_SUBS_PROF 528
 IBMQREP_TABVERSION 529
 IBMQREP_TARGETS 555
 IBMQREP_TRG_COLS 563
 tables cible
 autorisations nécessaires 4
 CCD 212
 clé cible
 abonnements Q 87
 publications 230

tables cible (*suite*)
 colonnes de clé
 abonnements Q 87
 publications 230
 colonnes de l'index, abonnements
 Q 87
 dans les abonnements Q
 conditions inattendues 89
 exigences de connectivité 6
 gestion 338
 IBMQREP_TARGETS, table de
 contrôle 530
 mappage vers les sources
 réplication Q 78
 options de chargement
 aucun chargement 191, 199
 automatique 192
 chargement automatique 191
 chargement manuel 191
 présentation 191
 recommandations 191
 réplication bidirectionnelle 199
 réplication entre homologues 199
 réplication unidirectionnelle 199
 pour une réplication
 bidirectionnelle 97
 pour une réplication entre
 homologues 99
 pour une réplication unidirectionnelle
 conditions inattendues 89
 description 61
 profils objet cible 67
 règles d'attribution de nom 369
 réparation 350
 Table de contrôle
 IBMQREP_TARGETS 555
 tables de contrôle 319, 522
 analyse 475
 création pour Oracle 170
 élagage 335
 gestion 335
 publication d'événements
 présentation 489
 Q Apply
 suppression 311
 Q Apply, serveur
 liste 530
 Q Capture
 suppression 311, 319
 Q Capture, serveur
 IBMQREP_EXCLSHEMA 505
 liste 489
 réorganisation 337
 réplication Q
 présentation 489
 Serveur Q Apply
 IBMQREP_APPENVINFO 532
 IBMQREP_APPLYENQ 531
 IBMQREP_APPLYMON 533
 IBMQREP_APPLYPARMS 538
 IBMQREP_APPLYTRACE 545
 IBMQREP_DELTOMB 546
 IBMQREP_DONEMSG 547
 IBMQREP_EXCEPTIONS 547
 IBMQREP_RECVQUEUEES 550
 IBMQREP_SAVERI 552
 IBMQREP_SPILLEDROW 554

- tables de contrôle *(suite)*
 - Serveur Q Apply *(suite)*
 - IBMQREP_SPILLQS 554
 - IBMQREP_TARGETS 555
 - IBMQREP_TRG_COLS 563
 - serveur Q Capture 495
 - IBMQREP_ADMINMSG 490
 - IBMQREP_CAPENQ 491
 - IBMQREP_CAPENVINFO 492
 - IBMQREP_CAPMON 493
 - IBMQREP_CAPPARMS 495
 - IBMQREP_CAPQMON 500
 - IBMQREP_CAPTRACE 502
 - IBMQREP_COLVERSION 503
 - IBMQREP_IGNTRAN 506
 - IBMQREP_IGNTRANTRC 507
 - IBMQREP_PART_HIST 507
 - IBMQREP_SCHEMASUBS 509
 - IBMQREP_SENDQUEUES 511
 - IBMQREP_SIGNAL 515
 - IBMQREP_SRC_COLS 520
 - IBMQREP_SRCH_COND 521
 - IBMQREP_SUBS 522
 - IBMQREP_SUBS_PROF 528
 - IBMQREP_TABVERSION 529
 - serveur Q Capture pour Oracle
 - IBMQREP_EOLFLUSH 504
 - suppression 311, 319
 - utilitaire RUNSTATS 337
- tables de contrôle, création 52
- tables de contrôle, création sur une version différente 54
- tables de contrôle Q Apply
 - création 52
 - suppression 311
- Tables de contrôle Q Capture
 - création 52
 - Oracle 170
 - suppression 311, 319
- tables de modification cohérente des données (CCD) 212
- tables logiques
 - réplication bidirectionnelle 97
 - réplication entre homologues 99
- tables partitionnées
 - réplication 209, 211
- tables partitionnées par intervalle
 - réplication 209, 211
- tables source
 - ajout de colonnes à répliquer
 - publications existantes 315
 - réplication bidirectionnelle ou entre homologues. 144, 298
 - réplication unidirectionnelle 141, 295
 - dans des publications
 - clauses WHERE 227
 - conditions de recherche 227
 - filtrage de lignes 227
 - lors de l'envoi d'un message par Q Capture 226
 - dans les abonnements Q
 - conditions de recherche 73
 - filtrage de lignes 73
 - mappage des colonnes source et cible 78
 - messages Q Capture 72
- tables source *(suite)*
 - gestion 331
 - mappage vers des procédures stockées 188
 - mappage vers les cibles, réplication Q 78
 - modification dans Oracle 177
 - modifications de structure 129, 300
 - pour une réplication bidirectionnelle 97
 - pour une réplication entre homologues 99
 - pour une réplication unidirectionnelle
 - description 61
 - règles d'attribution de nom 369
 - réplication de chargements 203, 205
 - tâches démarrées 361
 - temps d'attente
 - présentation 326
 - temps d'attente de file d'attente 326
 - temps d'attente de transaction
 - Q Capture 326
 - temps d'attente de transaction Q Capture 326
 - temps d'attente Q Apply 326
 - temps d'attente Q Capture 326
 - TIMESTAMP WITH TIMEZONE 233
 - Tivoli NetView pour z/OS, utilisation avec la réplication Q 41
 - transactions
 - ignorer
 - Q Apply 289
 - Q Capture 259
 - messages 571
 - transformation de données
 - procédures stockées pour les abonnements Q
 - présentation 181
 - réplication Q
 - procédures stockées, description 181
 - Réplication Q
 - procédures stockées, présentation 181
 - transformation des données 80, 84
 - procédures stockées, description 181
 - procédures stockées, présentation 181
 - type de données BLOB
 - message LOB 582
 - réplication 235
 - valeurs de colonne 582
 - type de données CLOB
 - message LOB 582
 - réplication 235
 - valeurs de colonne 582
 - Type de données DATE
 - réplication 239
 - Type de données DBCLOB
 - message LOB 582
 - réplication 235
 - valeurs de colonne 582
 - type de données LOB
 - longueur de file d'attente 35
 - message LOB 582
 - messages 582
 - réplication 235
- type de données LOB *(suite)*
 - restrictions pour une réplication bidirectionnelle 110
 - valeurs de colonne 582
- type de données LONG VARCHAR
 - présentation 233
- type de données LONG VARCHARIC
 - présentation 233
- Type de données NUMBER
 - réplication 239
- Type de données TIMESTAMP
 - réplication 239
- type de données XML
 - publication 237
 - réplication 237
 - restrictions avec des données d'objet LOB 238
- types de données
 - LOB, réplication 235
 - objet LOB
 - restrictions avec des données XML 238
- remarques
 - lors de la publication 233
 - lors de la réplication 233
- réplication
 - BLOB 235
 - CLOB 235
 - DBCLOB 235
 - restrictions 233
 - XML
 - réplication 237
 - restrictions avec des données d'objet LOB 238
- types de données abstraits
 - description 233
- types de données d'objet LOB
 - restrictions avec des données XML 238
- types de données distincts 233
- types UDT
 - réplication Q 233

U

- unités d'exécution
 - réplication Q
 - administration 323
 - agent 323
 - détention de verrou 323
 - élagage 323
 - navigateur 323
 - sérialisation 323
 - vérification du statut 323
 - unités d'exécution administration 323
 - unités d'exécution agent 323
 - programmes de réplication 323
 - taille des messages 34
 - temps d'attente 326
 - unités d'exécution d'administration 323
 - unités d'exécution d'élagage 323
 - unités d'exécution de détention de verrou 323
 - unités d'exécution du navigateur
 - programmes de réplication 323

- utilitaire asntdiff
 - avec fonctions de compatibilité DB2 350
 - présentation 341
- utilitaire asntrep
 - guide d'utilisation 350
- utilitaire asntreppair
 - guide d'utilisation 341
 - présentation 341
- utilitaire de comparaison des tables
 - exécution en mode parallèle 347
- utilitaire de différences des tables 341, 450, 454, 462
- utilitaire de différenciation des tables
 - avec fonctions de compatibilité DB2 350
 - type de données DATE
 - réplication 350
 - type de données NUMBER
 - réplication 350
 - type de données TIMESTAMP
 - réplication 350
- utilitaire de réparation des tables 350, 473
- utilitaire LOAD à partir de CURSOR 192, 195
- utilitaire RUNSTATS
 - réplication Q 337
- utilitaires
 - Analyseur de réplication Q 475
 - comparaison de tables 347
 - différenciation des tables 450, 454, 462
 - EXPORT et IMPORT 192
 - EXPORT et LOAD 192
 - formateur de messages 484, 486
 - LOAD à partir de CURSOR 192
 - réparation de table 473
- Utilitaires EXPORT et IMPORT 192

V

- valeurs après
 - incluses dans les publications 231
 - paramètres pour la clé 188
 - pour des colonnes de clé 181
 - répondre aux conditions de recherche 73
- valeurs avant
 - colonnes de non clé 188
 - colonnes ROWID 235
 - incluses dans les publications 231
 - procédures stockées 181
 - réplication bidirectionnelle, détection de conflit 110
 - restrictions pour les types de données LOB 110
 - Table IBMQREP_SUBS 522
 - type de données XML 237
 - types de données LOB 235
- variable ASNUSEMQCLIENT 26
- variable d'environnement
 - DB2CODEPAGE 43
- variable d'environnement
 - DB2ENVLIST 26
- variable d'environnement DB2INSTANCE
 - définition 43

- variable d'environnement EXTSHM 43
 - mémoire 45
- variable d'environnement TMPDIR 44
- variable LDR_CNTRL 45
- variable MQCHLLIB 26
- variable MQCHLTAB 26
- variable MQSERVER 26
- variables d'environnement
 - réglage des paramètres pour Linux, UNIX, et Windows 43
 - serveur fédéré 153
- vue V\$LOGMNR_CONTENTS 168

W

- WebSphere MQ
 - autorisations nécessaires 32
 - moniteur d'alertes de réplication 32
 - programme Q Apply 32
 - Q Capture, programme 32
 - besoins en mémoire 33
 - configuration de la publication d'événements 9
 - configuration de la réplication Q 9
 - contrôles de validation 29
 - limite de taille des messages
 - Q Apply 34
 - Q Capture 34
 - objets
 - autorisations nécessaires 32
 - exigences de connectivité 32
 - file d'attente d'administration 23
 - file d'attente d'envoi 23
 - file d'attente de déversement 23
 - file d'attente de réception 23
 - file d'attente de redémarrage 23
 - liste de 23
 - paramètres requis 18
 - publication d'événement, à distance 13
 - réplication bidirectionnelle 14
 - réplication entre homologues (deux serveurs distants) 14
 - réplication entre homologues (trois serveurs distants ou plus) 16
 - réplication unidirectionnelle, distante 9
 - réplication unidirectionnelle, même système 11
 - validation 27
 - paramètres
 - CHLTYPE 18
 - DEFPSIST 18
 - DEFSOPT 18
 - DEFTYPE 18
 - DISCINT 18
 - HBINT 18
 - MAXDEPTH 18
 - MAXMSGL 18
 - MSGDLVSQ 18
 - SHARE 18
 - segmentation des messages 34
 - validation d'objets 27

X

- XML, expressions 84

Z

- z/OS
 - remarques relatives aux chargements 194
 - spécification de CAPTURE_PATH 355



SC11-7048-00



Spine information:

IBM InfoSphere Data Replication

Version 10.1.3

Référence et guide pour la réplcation Q et la publication d'événements

