

IBM InfoSphere Data Replication  
Version 10.1.3

*Guide de référence de la  
réplication SQL*





IBM InfoSphere Data Replication  
Version 10.1.3

*Guide de référence de la  
réplication SQL*



**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 505.

**Première édition - mai 2012**

Réf. US : SC19-3638-00

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 1994, 2012.

# Table des matières

## Avis aux lecteurs canadiens. . . . . ix

## Chapitre 1. Planification pour la réplication SQL . . . . . 1

Planification de la migration . . . . .	1
Planification de la mémoire . . . . .	1
Mémoire utilisée par le programme Capture. . . . .	1
Mémoire utilisée par le programme Apply . . . . .	3
Planification du stockage . . . . .	3
Impact du journal pour les serveurs source DB2 . . . . .	3
Impact du journal pour les serveurs cible. . . . .	4
Besoins en mémoire des tables cible et des tables de contrôle . . . . .	5
Espace requis pour les fichiers auxiliaires du programme Capture . . . . .	6
Espace requis pour les fichiers auxiliaires du programme Apply . . . . .	7
Besoins en espace pour les fichiers journaux de diagnostique (z/OS, Linux, UNIX, Windows) . . . . .	8
Planification de la détection de conflit . . . . .	8
Planification d'une source relationnelle non DB2 . . . . .	9
Débits des transactions pour les déclencheurs de capture . . . . .	9
Impact du journal pour des serveurs source relationnels non DB2 . . . . .	9
Coexistence de déclencheurs existants avec des déclencheurs de capture . . . . .	9
Verrous pour des serveurs source Oracle . . . . .	10
Modifications nécessaires pour déclencheurs Sybase sur des plateformes little-endian . . . . .	10
Planification de la conversion de pages de codes. . . . .	10
Réplication entre des bases de données avec des pages de codes compatibles . . . . .	10
Pages de code pour la réplication SQL . . . . .	11
Planification de la réplication pour DB2 for z/OS. . . . .	12
Réglages des performances . . . . .	12

## Chapitre 2. Autorisations requises pour la réplication SQL . . . . . 15

Authentification requise sous Linux, UNIX et Windows . . . . .	15
Autorisations nécessaires pour l'administration . . . . .	15
Autorisations nécessaires pour le programme Capture . . . . .	17
Autorisations nécessaires pour le programme Apply . . . . .	18
Autorisations nécessaires pour les déclencheurs Capture sur les bases de données relationnelles non-DB2 . . . . .	19
Gestion des ID utilisateur et des mots de passe pour des serveurs distants (Linux, UNIX, Windows) . . . . .	20

## Chapitre 3. Configuration de serveurs pour la réplication SQL . . . . . 23

Obligatoire : Définition de DATA CAPTURE CHANGES sur les tables source DB2 et les tables système DB2 for z/OS. . . . .	23
Besoins de connectivité pour la réplication SQL . . . . .	24
Connexion aux serveurs System i à partir de Windows . . . . .	24
Connexion à des serveurs relationnels non DB2 . . . . .	25
Création de tables de contrôle pour la réplication SQL . . . . .	25
Création de tables de contrôle pour la réplication SQL . . . . .	26
Création de tables de contrôle (System i) . . . . .	27
Création de tables de contrôle pour des sources relationnelles non DB2. . . . .	27
Création de plusieurs ensembles de tables de contrôle Capture. . . . .	28
Création de tables de contrôle dans une base de données à partitions multiples . . . . .	29
Configuration des programmes de réplication . . . . .	29
Configuration des programmes de réplication (Linux, UNIX, Windows) . . . . .	29
Création de modules SQL à utiliser avec des systèmes distants (System i) . . . . .	32
Paramétrage des programmes de réplication (z/OS) . . . . .	34
Capture de plusieurs partitions de base de données . . . . .	34
Réplication de tables partitionnées : version 9.7 avec groupe de correctifs 1 ou précédent (Linux, UNIX, Windows) . . . . .	34
Réplication de tables partitionnées DB2 : version 9.7 avec groupe de correctifs 2 ou ultérieur (Linux, UNIX, Windows) . . . . .	36
Exécution de DB2 Query Patroller dans un environnement de réplication SQL. . . . .	38
Configuration des journaux (System i) . . . . .	38

## Chapitre 4. Enregistrement de tables et de vues en tant que sources de réplication SQL . . . . . 45

Enregistrement de tables DB2 en tant que sources . . . . .	45
Enregistrement de tables relationnelles non DB2 en tant que sources . . . . .	47
Options d'enregistrement applicables aux tables source . . . . .	49
Enregistrement d'un sous-ensemble de colonnes (vertical) . . . . .	49
Réplication en mode capture des modifications et régénération intégrale . . . . .	50
Colonnes d'image après et d'image avant . . . . .	51
Préfixe image avant . . . . .	54
Arrêt du programme Capture après une erreur . . . . .	54
Options de stockage des mises à jour par le programme Capture . . . . .	55

Comment éviter de recapturer les modifications (réplication bidirectionnelle uniquement) . . . . .	56
Options de détection de conflit (réplication bidirectionnelle) . . . . .	60
Enregistrement des tables qui utilisent la journalisation distante (System i) . . . . .	62
Intégrité référentielle sur la table cible lorsque la source est System i . . . . .	63
Utilisation de numéros relatifs d'enregistrement (RRN) à la place de clés primaires (System i) . . . . .	63
Comportement des vues en tant que sources de réplication . . . . .	64
Vues relatives à une seule table . . . . .	64
Vues relatives à la jonction de deux tables ou davantage . . . . .	65
Enregistrement de vues de tables en tant que sources . . . . .	67
Gestion de tables de modification cohérente des données (CCD) en tant que sources . . . . .	67

## Chapitre 5. Abonnement à des sources pour la réplication SQL . . . . . 69

Planification du mode de regroupement des sources et des cibles . . . . .	69
Planification du nombre de membres d'ensembles d'abonnements . . . . .	70
Planification du nombre d'ensembles d'abonnements par qualificatif Apply . . . . .	70
Création d'un ensemble d'abonnements . . . . .	71
Options de traitement pour les ensembles d'abonnements . . . . .	74
Indication si l'ensemble d'abonnements est actif . . . . .	74
Indication du nombre de minutes de données extraites par le programme Apply . . . . .	74
Options de chargement pour les tables cible avec intégrité référentielle . . . . .	76
Spécification de la réplication des modifications pour les membres d'ensembles d'abonnements par le programme Apply . . . . .	77
Définition d'instructions SQL ou de procédures stockées pour l'ensemble d'abonnements . . . . .	78
Options de planification de la réplication d'ensembles d'abonnements . . . . .	78
Planification de l'ensemble d'abonnements . . . . .	80
Création de membres d'ensembles d'abonnements . . . . .	81
Types de table cible . . . . .	83
Propriétés communes à tous les types de tables cible . . . . .	96

## Chapitre 6. Réplication de types spéciaux de données dans la réplication SQL . . . . . 103

Restrictions générales de données pour la réplication SQL . . . . .	103
Types de données d'objets LOB . . . . .	104
Réplication de nouveaux types de données DB2 version 9.7 (Linux, UNIX, Windows) . . . . .	105
Réplication de tables avec des colonnes d'identité . . . . .	107

## Chapitre 7. Sous-ensemble de données dans un environnement de réplication SQL . . . . . 109

Etablissement de sous-ensembles de données pendant l'enregistrement . . . . .	109
Création de sous-ensembles de données source à l'aide de vues . . . . .	110
Définition des déclencheurs dans des tables CD pour empêcher la capture de lignes spécifiques . . . . .	110
Définition de sous-ensembles de données pendant l'abonnement . . . . .	111

## Chapitre 8. Manipulation de données dans un environnement de réplication SQL . . . . . 113

Extension des données à l'aide de procédures stockées ou d'instructions SQL . . . . .	114
Mappage des colonnes source et cible portant des noms différents . . . . .	115
Création de colonnes calculées . . . . .	115

## Chapitre 9. Utilisation du programme Capture pour la réplication SQL . . . . . 117

Démarrage du programme Capture (Linux, UNIX, Windows et z/OS) . . . . .	117
Démarrage du programme Capture à partir d'un emplacement connu dans le journal DB2 . . . . .	119
Démarrage du programme Capture (System i) . . . . .	120
Paramètres d'exploitation par défaut du programme Capture . . . . .	120
Description des paramètres d'exploitation Capture . . . . .	122
Méthodes de modification des paramètres Capture . . . . .	133
Modification du comportement d'un programme Capture en cours d'exécution . . . . .	135
Modification des paramètres d'exploitation enregistrés dans la table IBMSNAP_CAPPARMS . . . . .	136
Arrêt du programme Capture . . . . .	137
Réinitialisation de Capture . . . . .	138
Interruption du programme Capture (Linux, UNIX, Windows, z/OS) . . . . .	138
Reprise de Capture (Linux, UNIX, Windows, z/OS) . . . . .	139

## Chapitre 10. Utilisation du programme Apply pour la réplication SQL . . . . . 141

Démarrage du programme Apply (Linux, UNIX, Windows, z/OS) . . . . .	141
Démarrage d'un programme Apply (System i) . . . . .	143
Paramètres d'exploitation par défaut du programme Apply . . . . .	144
Description des paramètres d'exploitation Apply . . . . .	146
Méthodes de modification des paramètres d'exploitation Apply . . . . .	155
Modification des paramètres Apply enregistrés dans la table IBMSNAP_APPPARMS (Linux, UNIX, Windows, z/OS) . . . . .	155
Arrêt du programme Apply . . . . .	156
Emplacement de stockage des détails relatifs aux échecs d'intégrité référentielle par le programme Apply . . . . .	156

Modification de la routine d'exit ASNDONE (Linux, UNIX, Windows, z/OS) . . . . .	157
Modification de la routine d'exit ASNDONE (System i) . . . . .	158
Actualisation des tables cible à l'aide de la routine d'exit ASNLOAD . . . . .	159
Actualisation des tables cible à l'aide de la routine d'exit ASNLOAD (Linux, UNIX, Windows) . . . . .	160
Actualisation des tables cible à l'aide de la routine d'exit ASNLOAD (z/OS) . . . . .	162
Personnalisation du comportement de l'exit ASNLOAD (Linux, UNIX, Windows, z/OS) . . . . .	163
Régénération des tables cible avec la routine d'exit ASNLOAD (System i) . . . . .	165
Actualisation d'une table dans un ensemble d'abonnements à plusieurs tables. . . . .	166
Veiller à ce que les utilitaires utilisés pour la régénération intégrale attendent les données validées . . . . .	167

## Chapitre 11. Utilisation des programmes de réplication (z/OS) . . . 169

Utilisation de tâches démarrées par le système pour utiliser les programmes de réplication . . . . .	169
Utilisation du langage JCL pour faire fonctionner les programmes de réplication. . . . .	169
Démarrage du programme Apply sur z/OS avec JCL. . . . .	171
Gérer les programmes de réplication SQL en cours d'exécution grâce à l'utilisation de la commande MVS MODIFY . . . . .	171
Démarrage du programme Capture sur z/OS avec JCL. . . . .	173
Utilisation du gestionnaire ARM (Automatic Restart Manager) pour redémarrer automatiquement la réplication et la publication (z/OS). . . . .	174
Migration de votre environnement de réplication en mode de partage de données (z/OS) . . . . .	175

## Chapitre 12. Modification d'un environnement de réplication SQL . . . 177

Enregistrement de nouveaux objets . . . . .	177
Modification des attributs d'enregistrement d'objets enregistrés . . . . .	178
Ajout de colonnes aux tables source. . . . .	178
Traitement des opérations ALTER TABLE ALTER COLUMN SET DATA TYPE . . . . .	180
Arrêt de la capture des modifications des objets enregistrés . . . . .	181
Enregistrements admissibles pour la réactivation	182
Suppression d'enregistrements. . . . .	183
Modification des schémas Capture . . . . .	184
Création de nouveaux ensembles d'abonnements	186
Ajout de membres d'ensembles d'abonnements à des ensembles existants . . . . .	186
Désactivation de membres d'ensembles d'abonnements existants. . . . .	187

Activation de membres d'ensembles d'abonnements dans des ensembles d'abonnements existants . . . . .	188
Modification des propriétés d'ensembles d'abonnements . . . . .	188
Modification des noms d'ensembles d'abonnements	189
Division d'un ensemble d'abonnements. . . . .	191
Fusion d'ensembles d'abonnements . . . . .	194
Modification des qualificatifs Apply des ensembles d'abonnements . . . . .	196
Désactivation d'ensembles d'abonnements. . . . .	198
Suppression d'ensembles d'abonnements . . . . .	200
Coordination entre les événements de réplication et les événements d'applications de base de données . . . . .	201
Définition d'un événement END_SYNCHPOINT à l'aide du signal USER . . . . .	201
Utilisation du signal Capture CMD STOP . . . . .	202
Exécution d'un signal d'établissement de liaison CAPSTART hors du programme Apply. . . . .	205
Exécution d'un signal CAPSTOP . . . . .	206
Réglage de l'heure d'été/d'hiver (System i) . . . . .	207
Options de promotion de votre configuration de réplication dans un autre système . . . . .	208

## Chapitre 13. Gestion d'un environnement de réplication SQL . . . 211

Gestion de systèmes source. . . . .	211
Accès aux tables et aux vues source . . . . .	211
Journaux source et récepteurs de journal . . . . .	211
Maintenance des tables de contrôle . . . . .	215
Utilitaire RUNSTATS pour la réplication SQL (Linux, UNIX, Windows, z/OS) . . . . .	215
Lier à nouveau les modules et plans (z/OS, Linux, UNIX, Windows). . . . .	216
Réorganisation des tables de contrôle . . . . .	216
Elagage des tables de contrôle dynamiques gérées par les programmes Capture (Linux, UNIX, Windows, z/OS) . . . . .	217
Elagage des tables de modification des données (CD) et des tables d'unités de travail (UOW) . . . . .	218
Recommandations relatives à l'élagage d'autres tables de contrôle dynamiques . . . . .	219
Prévention des échecs de réplication et reprise après erreur . . . . .	219
Gestion des tables cible . . . . .	222

## Chapitre 14. Comparaison et réparation de tables . . . . . 223

Utilitaire de comparaison de tables (asntdiff) . . . . .	223
Exécution de l'utilitaire asntdiff en mode parallèle (sous z/OS) . . . . .	229
Utilitaire de réparation des tables (asntrep) . . . . .	232
Gestion des fonctions de compatibilité SQL DB2 par l'utilitaire de comparaison. . . . .	232

## Chapitre 15. Planification de programmes de réplication SQL sur différents systèmes d'exploitation . . . 237

Planification de programmes sous Linux et UNIX	237
Planification de programmes sous Windows . . . . .	237

Planification des programmes sur les systèmes d'exploitation z/OS . . . . .	238
Planification des programmes sur le système d'exploitation System i . . . . .	238

**Chapitre 16. Services de réplication (Windows). . . . . 239**

Description des services Windows pour la réplication . . . . .	239
Création d'un service de réplication . . . . .	240
Lancement d'un service de réplication . . . . .	241
Arrêt d'un service de réplication . . . . .	241
Affichage d'une liste des services de réplication . . . . .	241
Suppression d'un service de réplication. . . . .	241

**Chapitre 17. Communication entre les composants de la réplication SQL . . . 243**

Centre de réplication, ASNCLP, programme ou déclencheurs Capture et programme Apply . . . . .	243
Programme Capture et programme Apply. . . . .	244
Déclencheurs Capture et programme Apply . . . . .	246
Outils d'administration et moniteur d'alertes de réplication . . . . .	247
Moniteur d'alertes de réplication, programme Capture et programme Apply . . . . .	247

**Chapitre 18. Vérification du statut des programmes de réplication SQL . . . 249**

Vérification du statut des programmes de réplication (z/OS, Linux, UNIX, Windows) . . . . .	249
Vérification du statut des travaux de journal Capture et Apply (System i) . . . . .	250
Suivi de l'avancement du programme Capture (System i) . . . . .	251

**Chapitre 19. Personnalisation et exécution de scripts SQL pour la réplication. . . . . 253**

**Chapitre 20. Règles d'attribution de nom applicables aux objets de la réplication SQL . . . . . 255**

**Chapitre 21. Commandes système pour la réplication SQL (Linux, UNIX, Windows, z/OS) . . . . . 257**

asnccmd : démarrage de Capture . . . . .	257
asnccmd : fonctionnement de Capture . . . . .	266
asnapply : démarrage du programme Apply . . . . .	270
asnacmd : Apply en fonctionnement . . . . .	277
asnanalyze : fonctionnement de l'analyseur . . . . .	278
asnpwd : création et gestion des fichiers de mots de passe . . . . .	281
ansrcrt : création d'un service de réplication . . . . .	285
ansndrop : suppression d'un service de réplication . . . . .	288
asnslst : liste des services de réplication . . . . .	289
asntdiff : comparaison des données dans les tables source et cible (Linux, UNIX, Windows) . . . . .	290

asntdiff : comparaison des données dans les tables source et cible (z/OS) . . . . .	294
Option de commande asntdiff -f (fichier en entrée) . . . . .	301
asntrc : utilisation de la fonction trace de réplication . . . . .	305
asntrep : réparation des différences entre les tables source et cible . . . . .	313

**Chapitre 22. Commandes systèmes pour la réplication SQL (System i) . . 317**

ADDDPRREG : ajout d'un enregistrement DPR (System i) . . . . .	317
ADDDPRSUB : ajout d'un ensemble d'abonnements DPR (System i) . . . . .	326
ADDDPRSUBM : ajout d'un membre d'un ensemble d'abonnements DPR (System i) . . . . .	341
ANZDPR : fonctionnement de l'Analyseur (System i) . . . . .	352
CHGDPRCAPA : modification des attributs DPR Capture (System i). . . . .	355
CRTDPRtbl : création de tables de contrôle de réplication (System i). . . . .	359
ENDDPRAPY : arrêt de Apply (System i) . . . . .	360
ENDDPRCAP : arrêt de Capture (System i) . . . . .	363
GRTDPRAUT : autorisation des utilisateurs (System i) . . . . .	365
INZDPRCAP : réinitialisation de la capture DPR (System i) . . . . .	374
OVRDPRCAPA : substitution des attributs DPR Capture (System i). . . . .	375
RMVDPRREG : suppression d'un enregistrement DPR (System i). . . . .	380
RMVDPRSUB : suppression d'un ensemble d'abonnements DPR (System i) . . . . .	381
RMVDPRSUBM : suppression d'un membre d'un ensemble d'abonnements DPR (System i) . . . . .	383
RVKDPRAUT : révocation des droits d'accès(System i) . . . . .	384
STRDPRAPY : démarrage de Apply (System i) . . . . .	386
STRDPRCAP : démarrage de Capture (System i) . . . . .	393
WRKDPTRC : utilisation de la fonction trace DPR (System i) . . . . .	401

**Chapitre 23. Structures des tables de réplication SQL . . . . . 407**

Tables d'aperçu rapide . . . . .	407
Tables sur le serveur de contrôle de Capture . . . . .	414
Table IBMSNAP_AUTHTKN (System i) . . . . .	416
Table IBMSNAP_CAPENQ (z/OS, Linux, UNIX, Windows) . . . . .	417
Table IBMSNAP_CAPMON . . . . .	418
Table IBMSNAP_CAPPARMS . . . . .	420
Table IBMSNAP_CAPSCHEMAS . . . . .	424
Table IBMQREP_COLVERSION . . . . .	425
Table IBMSNAP_CAPTRACE . . . . .	426
Table de modification des données (non DB2) . . . . .	427
Table CD . . . . .	428
Table IBMQREP_IGNTRAN . . . . .	429
Table IBMQREP_IGNTRANTRC . . . . .	430
Table IBMSNAP_PARTITIONINFO . . . . .	431



Table IBMSNAP_PRUNCNTL . . . . .	432
Table IBMSNAP_PRUNE_LOCK . . . . .	434
Table IBMSNAP_PRUNE_SET . . . . .	435
IBMSNAP_REG_EXT (System i) . . . . .	435
Table IBMSNAP_REGISTER . . . . .	437
Table IBMSNAP_REG_SYNCH (relationnelle non DB2) . . . . .	444
Table IBMSNAP_RESTART . . . . .	444
Table IBMSNAP_SEQTABLE (Informix) . . . . .	447
Table IBMSNAP_SIGNAL . . . . .	447
Table IBMQREP_TABVERSION . . . . .	450
Table IBMSNAP_UOW . . . . .	451
Tables sur le serveur de contrôle Apply . . . . .	453
Table ASN.IBMSNAP_APPENQ . . . . .	454
Table ASN.IBMSNAP_APPLEVEL . . . . .	455
ASN.IBMSNAP_APPLY_JOB (System i) . . . . .	455
Table ASN.IBMSNAP_APPLYMON . . . . .	456
Table ASN.IBMSNAP_APPPARMS . . . . .	457
Table ASN.IBMSNAP_APPLYTRACE . . . . .	461
Table ASN.IBMSNAP_APPLYTRAIL . . . . .	462
Table ASN.IBMSNAP_FEEDETL . . . . .	468
Table ASN.IBMSNAP_SUBS_COLS . . . . .	469
Table ASN.IBMSNAP_SUBS_EVENT . . . . .	471
Table ASN.IBMSNAP_SUBS_MEMBR . . . . .	472
Table ASN.IBMSNAP_SUBS_SET . . . . .	476
Table ASN.IBMSNAP_SUBS_STMTS . . . . .	483
Tables sur le serveur de contrôle cible . . . . .	485
Table d'agrégation de base . . . . .	485
Table d'agrégation des modifications . . . . .	486
Cibles CCD . . . . .	486
Table des points de cohérence . . . . .	489

Table réplique . . . . .	489
Table de copie utilisateur . . . . .	490

**Annexe A. Schémas de codage  
UNICODE et ASCII pour la réplication  
SQL (z/OS) . . . . . 493**

Règles pour la sélection d'un schéma de codage . . . . .	493
Paramétrage des schémas de codage . . . . .	494

**Annexe B. Démarrage des  
programmes de réplication SQL  
depuis une application (Linux, UNIX,  
Windows) . . . . . 495**

**Annexe C. Comment le programme  
Capture traite les types d'entrée de  
journal pour la réplication SQL  
(System i) . . . . . 497**

**Contacteur IBM . . . . . 501**

**Lecture des diagrammes de syntaxe 503**

**Remarques et marques . . . . . 505**

Marques . . . . .	508
-------------------	-----

**Index . . . . . 511**



---

## 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. Planification pour la réplication SQL

Lors de la planification de la réplication SQL, il se peut que vous deviez envisager la planification de la migration, de la mémoire, du stockage, des conflits, des systèmes source, de la conversion de page de code, et des performances.

---

## Planification de la migration

La planification de la migration suppose celle de problèmes potentiels lors de la migration d'une version de réplication à une autre.

Si vous migrez d'un environnement de réplication à un autre, certains problèmes de migration doivent être pris en compte. *WebSphere Information Integration Migrating to Replication Version 9* décrit comment migrer vers la réplication version 9. Pour migrer vers la version 9, vos serveurs doivent d'abord se trouver à la version 8. *WebSphere Information Integration Migrating to SQL Replication Version 8* décrit comment migrer vers la réplication version 8. Ce document décrit également comme migrer des environnements de réplication utilisant actuellement DB2 DataJoiner pour répliquer des données de ou vers des serveurs relationnels non DB2. Ces documents sont disponibles en ligne sur le site de support de WebSphere Information Integration pour votre produit.

---

## Planification de la mémoire

La planification de la mémoire implique celle de la quantité de mémoire requise par la réplication. Celle-ci utilise uniquement la mémoire nécessaire. Cette quantité est directement proportionnelle à celle des données répliquées depuis la source et l'accès concurrent des transactions. Plus il y a de données répliquées, plus il y a de transactions simultanées et plus l'application requiert de la mémoire.

L'exécution des programmes Capture et Apply peut solliciter une quantité importante de mémoire.

### Mémoire utilisée par le programme Capture

Le programme Capture utilise de la mémoire lorsqu'il lit le journal de récupération DB2. Il stocke en mémoire des enregistrements de transaction individuels jusqu'à ce qu'il ait lu l'enregistrement de validation ou d'abandon associé. Les données associées à une transaction abandonnée sont effacées de la mémoire, alors que celles associées à un enregistrement de validation sont écrites dans la table CD ou UOW. Les transactions validées restent en mémoire tant que le programme Capture ne valide pas son travail lorsqu'il parvient à son intervalle de validation.

Pour contrôler la quantité de mémoire utilisée par le programme Capture, observez la colonne CURRENT\_MEMORY de la table IBMSNAP\_CAPMON.

Vous pouvez définir le paramètre **memory\_limit** au démarrage du programme Capture afin que ce dernier utilise une quantité indiquée de mémoire pour le stockage associé aux transactions. Les autres utilisations de la mémoire ne sont pas limitées par ce paramètre. Vous pouvez aussi modifier le paramètre **memory\_limit** pendant l'exécution du programme Capture. Si celui-ci atteint la limite de

mémoire, il écrit des transactions dans un fichier auxiliaire. Vous devez évaluer les ressources mémoire utilisées par le programme Capture en fonction des besoins d'espace de stockage.

Vous pouvez aussi prendre en compte la taille des transactions utilisateur et l'intervalle de validation au moment de planifier les besoins en mémoire du programme Capture. Les travaux par lots d'exécution longue dans l'intervalle de validation sollicitent beaucoup de mémoire lors de l'exécution du programme Capture. En général, plus l'intervalle de validation est petit, moins le programme Capture requiert de mémoire.

Les informations sur les enregistrements actifs sont lues et stockées en mémoire lorsque vous démarrez une instance du programme Capture et ajoutez de façon dynamique des enregistrements pendant l'exécution du programme Capture.

#### z/OS

#### Linux UNIX Windows

Lorsque la réplication lit des enregistrements de journal, elle utilise une mémoire tampon. Sa taille par défaut sur le système d'exploitation z/OS est de 66 pages de 1 ko et il s'agit d'une mémoire ECSA (zone système commune étendue). La réplication utilise uniquement la zone système commune étendue dans ce cas. La taille par défaut de cette mémoire tampon sur les systèmes d'exploitation Linux, UNIX et Windows est de 50 pages de 4 ko.

#### System i

CURRENT\_MEMORY est la quantité actuelle de mémoire supplémentaire allouée pour conserver les enregistrements de transactions au-delà de la mémoire utilisée par les mémoires tampons d'entrée-sortie standard pour les tables CD actives. Vous connaissez ainsi la quantité de mémoire supplémentaire actuellement utilisée pour conserver un grand nombre de transactions. Il ne s'agit pas de la quantité exacte de la mémoire globale utilisée par le travail de consignment spécifique.

Les informations stockées dans la table IBMSNAP\_CAPMON offrent des statistiques opérationnelles pour ajuster l'utilisation de la mémoire. Les valeurs dans cette table concernent une fréquence de contrôle de Capture déterminée et ne se cumulent pas entre les fréquences de contrôle. Les données dans la colonne CURRENT\_MEMORY ne contiennent pas d'autre quantité. Elles reflètent la mémoire utilisée à la fin de la fréquence de contrôle après la création de l'enregistrement. La fréquence de contrôle de Capture détermine à quelle fréquence le programme Capture insère des données dans cette table. Utilisez l'une des méthodes suivantes pour ajuster la quantité de mémoire utilisée par le programme Capture :

#### **Ajustement de la limite de mémoire pour permettre des déversements :**

1. Au démarrage du programme Capture, utilisez la limite de mémoire par défaut.
2. Vérifiez si des données sont déversées de la mémoire dans un fichier temporaire en observant la colonne TRANS\_SPILLED de la table IBMSNAP\_CAPMON. Cette colonne montre le nombre de transactions système source déversées sur le disque en raison de restrictions de mémoire lors d'une fréquence de contrôle déterminée de Capture.
3. Si des données sont déversées de la mémoire, prenez une limite supérieure ou un intervalle de validation moindre.

#### **Ajustement de la limite de mémoire pour éviter des déversements :**

1. Au démarrage du programme Capture, définissez une limite de mémoire élevée. La quantité dépend de vos ressources système.
2. Vérifiez la quantité de mémoire utilisée en observant la colonne CURRENT\_MEMORY de la table IBMSNAP\_CAPMON. Cette colonne montre la quantité de mémoire (en octets) utilisée par le programme Capture lors d'une fréquence de contrôle déterminée de Capture.
3. Si beaucoup moins de mémoire que celle indiquée pour la limite est utilisée, entrez une valeur inférieure.

## Mémoire utilisée par le programme Apply

Le programme Apply utilise de la mémoire lorsqu'il extrait des données. La quantité de mémoire utilisée est proportionnelle à la taille des colonnes de la table et au nombre de lignes extraites à la fois. Par exemple, si le programme Apply extrait une colonne LOB, il se peut qu'il utilise 2 Go de mémoire.

Les informations sur les ensembles d'abonnements actifs sont lues et stockées en mémoire lors de l'exécution du programme Apply. La quantité de mémoire utilisée par le programme Apply est généralement proportionnelle à celle de mémoire requise pour traiter l'ensemble d'abonnements possédant le plus de membres.

---

## Planification du stockage

La planification du stockage est importante pour l'impact du journal des serveurs source DB2, l'impact du journal des serveurs cible, les besoins d'espace des tables cible et des tables de contrôle, les besoins d'espace des fichiers journaux de diagnostic (Linux, UNIX, Windows, z/OS), les besoins d'espace des fichiers auxiliaires pour le programme Capture, ainsi que les besoins d'espace pour les fichiers auxiliaires du programme Apply.

Outre la mémoire requise pour DB2, vous devez vérifier la disponibilité de la mémoire pour la réplication pour les rubriques ci-après. Toutes les tailles indiquées dans ces rubriques sont seulement des estimations. Pour préparer et concevoir un système prêt pour la production, vous devez aussi prendre en compte des facteurs comme la prévention des incidents. Par exemple, la période de conservation des données doit éventuellement être augmentée pour gérer les indisponibilités réseau potentielles.

**Conseil :** Si les estimations de mémoire semblent trop élevées, observez à quelle fréquence le programme Apply exécute des ensembles d'abonnements et à quelle fréquence vos tables de réplication sont supprimées. Vous devez peut-être trouver un compromis entre l'utilisation de la mémoire, la capacité de tolérance des incidents et le temps système de l'unité centrale.

## Impact du journal pour les serveurs source DB2

En général, vous avez besoin de trois fois le volume actuel du journal pour toutes les tables impliquées dans une réplication. Il vous faut surtout de l'espace pour la table source ainsi que la table CD et les tables de contrôle de réplication. Cette section présente d'autres facteurs pouvant vous aider à évaluer l'impact du journal de façon plus précise que dans votre environnement de réplication.

Prenez en compte les mises à jour effectuées dans la base de données source par vos applications, ainsi que les besoins de réplication. Par exemple, si une application met à jour 60 % des colonnes d'une table, les besoins de réplication

peuvent faire grossir de plus de la moitié les enregistrements de journal par rapport à une table similaire non répliquée.

#### z/OS

#### Linux UNIX Windows

- DB2 journalise des images de ligne entière pour chaque instruction UPDATE. Cette opération se produit car, avant de pouvoir répliquer une table, vous devez la créer (ou la modifier) avec les mots clés DATA CAPTURE CHANGES.
- L'un des besoins de réplication ajoutant la majeure partie du journal est la capture d'images-avant et d'images-après (comme pour les tables réplique cible dans des scénarios de réplication bidirectionnelle). Une façon de réduire le volume du journal consiste à diminuer le nombre de colonnes définies pour la source de réplication. Par exemple, ne capturez pas d'images-avant si elles ne sont pas obligatoires.

#### System i

- DB2 journalise des images de ligne entière pour chaque instruction UPDATE. Une façon de réduire le volume du journal consiste à diminuer le nombre de colonnes définies pour la source de réplication ; par exemple, ne capturez pas d'images-avant si elles ne sont pas obligatoires.
- Pour réduire la quantité de mémoire utilisée pour des tables CD et UOW, réorganisez fréquemment ces tables, sachant que DASD ne fait pas de récupération automatique. Vous pouvez employer le mot clé RGZCTLTLBL (Reorganize Control Tables) dans la commande **ENDDPRCAP** pour réorganiser des tables de contrôle. Observez les modèles d'utilisation de DASD dans des conditions normales de fonctionnement pour prévoir et gérer l'utilisation de DASD. Si la consignation est activée, pensez aussi que le volume du journal augmente avec les insertions et les suppressions du journal DB2 des tables UOW et CD.
- Une fois le récepteur en cours plein, le système passe à un autre. Vous pouvez éventuellement sauvegarder et supprimer les anciens récepteurs devenus inutiles pour la réplication. Lorsqu'un système gère un grand nombre de transactions, le programme Capture peut parfois prendre du retard. Si le programme Capture est souvent en retard, vous pouvez distribuer vos tables source en plusieurs journaux afin de répartir la charge de travail entre diverses instances du programme Capture.

## Impact du journal pour les serveurs cible

Outre celle pour la base de données source, une consignation a lieu pour la base de données cible, dans laquelle les lignes sont appliquées. L'impact sur le journal dépend du mode de validation choisi pour le programme Apply.

### Mode table

Avec un traitement en mode table, le programme Apply effectue une seule validation une fois toutes les données extraites appliquées. Il n'émet pas de points de contrôle d'intervalle. Dans ce cas, vous devez évaluer la quantité maximum de données que le programme Apply traitera pendant un intervalle et modifier l'espace journal afin d'accueillir cette quantité de données.

### Mode transaction

Avec un traitement en mode transaction, le programme Apply copie dans les tables cible chaque mise à jour dans l'ordre des transactions et valide ces modifications à un intervalle sur une limite de transaction. Vous définissez l'intervalle pour les validations d'intervalle en définissant la



valeur de  $x$  dans l'option de l'ensemble d'abonnements **commit\_count**( $x$ ). Après que le programme Apply a extrait tous les ensembles de réponses, il applique le contenu des fichiers auxiliaires en suivant la séquence de validation. Ce type de traitement permet d'ouvrir et de traiter simultanément tous les fichiers auxiliaires. Par exemple, si vous prenez 1 comme nombre de validations, le programme Apply effectue une validation après chaque transaction ; si vous entrez 2 comme valeur, la validation a lieu après chaque ensemble de deux transactions.

**System i** Vous devez aussi prendre en compte l'espace journal (espace des récepteurs de journal) des tables cible. Comme les récepteurs de journal pour les tables cible sur System i peuvent être créés avec les paramètres **MNGRCV(\*SYSTEM)** et **DLTRCV(\*YES)** et sachant que vous devez journaliser uniquement les colonnes image-après, utilisez la formule suivante pour évaluer le volume de ces récepteurs :

```
volume_recepteurs_journal=longueur_ligne_table_cible  
X seuil_recepteurs_journal
```

## Besoins en mémoire des tables cible et des tables de contrôle

Vous devez évaluer le volume des nouvelles tables cible. L'espace requis pour une table cible ne dépasse en général pas celui d'une table source, mais il peut toutefois être plus important si la table cible est dénormalisée ou comporte des images-avant (en plus d'images-après) ou des données historiques. La taille de la table cible dépend de ce que vous choisissez de répliquer, comme le pourcentage de la table source que vous répliquez, le type de données des colonnes répliquées, si vous répliquez des images-avant et des images-après, si vous ajoutez des colonnes calculées, si vous définissez des sous-ensembles de lignes et si des transformations se produisent lors de la réplication.

Les tables CD et certaines tables de contrôle de réplication (IBMSNAP\_UOW, IBMSNAP\_CAPTRACE, IBMSNAP\_APPLYTRACE, IBMSNAP\_APPLYTRAIL, IBMSNAP\_CAPMON, IBMSNAP\_ALERTS) affectent également l'espace disque requis pour les bases de données source DB2. Ces tables peuvent grossir énormément en fonction de la configuration de votre environnement de réplication. L'espace requis pour les autres tables de contrôle de réplication est généralement faible et statique.

Les tables CD grossissent à chaque modification d'une table source tant que le programme Capture n'élague pas la table CD. Pour évaluer l'espace requis pour les tables CD, déterminez d'abord combien de temps vous souhaitez conserver les données avant leur élagage, puis indiquez à quelle fréquence le programme Capture doit automatiquement élaguer ces tables et à quelle fréquence vous élaguerez les tables à l'aide d'une commande.

Lorsque vous calculez le nombre d'octets de données répliquées, vous devez inclure 21 octets pour les données de temps système pour chaque ligne ajoutée aux tables CD par le programme Capture. Calculez la durée pendant laquelle le programme Capture doit pouvoir capturer des données dans des tables CD, même lorsque les données ne sont pas applicables, comme dans le cas d'une indisponibilité réseau. Évaluez le nombre d'insertions, de mises à jour et de suppressions qui seraient normalement capturées pour la table source dans la période de contingence.

Pour déterminer la taille conseillée pour la table CD, observez ce qui suit :

```
taille_CD_conseillée =  
  ( (21 octets) + sum(longueur de toutes les colonnes enregistrées) ) X  
  (nombre d'insertions, de mises à jour et de suppressions dans la table source  
  lors de la période de contingence)
```

## Exemple

Si les lignes dans la table CD ont une longueur de 100 octets (plus les 21 pour le temps système) et que 100 000 mises à jour sont capturées pendant une période de contingence de 24 heures, la mémoire requise pour la table CD est d'environ 12 Mo.

Les colonnes enregistrées dans cette formule comportent des colonnes d'images-avant et d'images-après. Si des mises à jour sont converties en paires d'opérations INSERT et DELETE, prenez-les en compte au moment de calculer le nombre d'insertions, de mises à jour et de suppressions. Par exemple, comptez chaque mise à jour de la table source comme deux lignes dans la table CD.

La table UOW grossit et se réduit en fonction du nombre de ligne insérées par le programme Capture lors d'un intervalle de validation déterminé et du nombre de lignes supprimées. Une ligne est insérée dans la table UOW chaque fois qu'une transaction d'application émet une opération COMMIT et que la transaction exécute une opération INSERT, DELETE ou UPDATE par rapport à une table source de réplication enregistrée. Vous devez d'abord surestimer l'espace requis par la table et contrôler l'espace réellement utilisé pour connaître celui à récupérer.

## Espace requis pour les fichiers auxiliaires du programme Capture

Si le programme Capture ne dispose pas d'assez de mémoire, il écrit les transactions dans des fichiers auxiliaires. Il écrit la transaction la plus importante dans un fichier, même s'il ne s'agit pas forcément de celle dépassant la limite de mémoire.

### z/OS

Les fichiers auxiliaires passent en entrée/sortie virtuelle (VIO).

### Linux UNIX Windows

Les fichiers auxiliaires sont toujours sur le disque. Un fichier par transaction est créé dans le répertoire **chemin\_capture**.

### System i

Les fichiers auxiliaires sont créés dans la bibliothèque QTEMP, un pour chaque enregistrement le demandant.

La taille des fichiers auxiliaires de Capture dépend des facteurs suivants :

### Taille de la mémoire

Utilisez le paramètre facultatif **limite\_mémoire** pour indiquer la quantité de mémoire utilisable par le programme Capture. Plus vous autorisez de mémoire, moins le programme Capture générera de fichiers auxiliaires.

### Taille des transactions

Des transactions volumineuses peuvent augmenter le besoin de fichiers auxiliaires.

### Nombre de transactions simultanées

Si le programme Capture traite plus de transactions à la fois ou s'il traite des transactions imbriquées, il doit stocker plus d'informations en mémoire ou sur le disque.

### Fréquence de validation

En général, plus l'intervalle de validation est réduit, plus le besoin en mémoire est faible car le programme Capture doit stocker moins longtemps en mémoire des informations avant de les valider.

## Espace requis pour les fichiers auxiliaires du programme Apply

Le programme Apply a besoin d'un espace temporaire pour stocker des données. (Si vous employez l'utilitaire ASNLOAD, vous pouvez avoir un fichier en entrée de chargement au lieu d'un fichier auxiliaire de chargement.) Le programme Apply utilise des fichiers auxiliaires pour conserver les mises à jour jusqu'à ce qu'il les applique aux tables cible. En général, les fichiers auxiliaires sont des fichiers disque ; toutefois, sur des systèmes d'exploitation z/OS, vous pouvez préciser que les données seront déversées dans la mémoire. Sauf en cas de contraintes de mémoire virtuelle, stockez les fichiers auxiliaires dans cette mémoire au lieu de les placer sur le disque.

La taille du fichier auxiliaire est proportionnelle à celle des données sélectionnées pour réplication au cours de chaque intervalle de réplication. En général, le fichier auxiliaire fait environ deux fois la taille des données. Vous pouvez évaluer la taille du fichier auxiliaire en comparant l'intervalle de fréquence (ou valeur de groupage de réplication) planifié pour le programme Apply au volume des modifications au cours de cette même période (ou dans une période d'heures pleines de modification).

**z/OS** **Linux UNIX Windows** La taille de ligne du fichier auxiliaire est celle de la *ligne cible*, y compris les colonnes de temps système de réplication. La ligne cible ne se trouve pas au format interne de DB2, mais dans un format alphanumérique interprété et développé (comme dans le cas d'une extraction de SELECT). La ligne comporte aussi une longueur et des caractères de fin NULL dans des chaînes individuelles. L'exemple suivant évalue la taille du fichier auxiliaire requis pour les données sélectionnées pour réplication et ne prend pas en compte l'espace supplémentaire requis pour les autres données stockées dans ce fichier.

**System i** La taille de ligne du fichier auxiliaire est toujours de 32 ko.

### Exemple

Si la modification de volume atteint 12 000 mises à jour par heure et que la fréquence du programme Apply est planifiée pour des intervalles d'une heure, le fichier auxiliaire doit conserver l'équivalent d'une heure de mises à jour, soit 12 000. Si chaque mise à jour représente 100 octets de données, le fichier auxiliaire fera au moins 1,2 Mo environ. L'espace supplémentaire est obligatoire pour les autres données stockées dans le fichier auxiliaire.

## Besoins en espace pour les fichiers journaux de diagnostic (z/OS, Linux, UNIX, Windows)

Les fichiers journaux de diagnostic stockent des informations sur les activités des programmes de réplication, comme le moment où le programme a démarré et s'est arrêté, ainsi que d'autres messages d'erreur et d'information provenant du programme. Par défaut, le programme ajoute des messages à son fichier journal, même après son redémarrage. Assurez-vous que les répertoires contenant ces fichiers journaux possèdent suffisamment d'espace pour stocker les fichiers.

L'emplacement des fichiers journaux de diagnostic dépend de la valeur définie pour les paramètres de démarrage **capture\_path**, **apply\_path** et **monitor\_path** au démarrage du programme Capture, du programme Apply et du programme de moniteur d'alertes de réplication.

Si la mémoire est un aspect important, vous pouvez réutiliser les journaux du programme afin que ce dernier supprime son journal et le recrée à chaque fois qu'il démarre. Vous pouvez donc indiquer si le journal doit être utilisé au démarrage du programme.

---

## Planification de la détection de conflit

Si vous utilisez la détection de conflit standard ou évoluée, vous devez stocker des images-avant dans les tables CD (ou CCD) pour les tables réplique cible. Par ailleurs, les règles d'intégrité référentielle sont restreintes. Dans des scénarios de réplication entre homologues ou bidirectionnelle, ou lorsque le programme Apply réalise un traitement en mode transaction, vous devez définir des règles d'intégrité référentielle compatibles avec les règles source. Dans le cas d'une réplication entre homologues ou bidirectionnelle, vous devez concevoir votre environnement d'application de façon à éviter des conflits de mise à jour si vous ne voulez pas activer la détection de conflit. Si des conflits sont possibles dans votre environnement d'application, vous pouvez sauvegarder des cycles de traitement en n'utilisant pas la détection de conflit.

Utilisez l'une des méthodes suivantes pour éviter des conflits dans une réplication entre homologues ou bidirectionnelle :

### Fragmentation par clé

Concevez votre application de façon à ce que la source de réplication soit mise à jour par des répliques pour les fourchettes de clés sur des sites spécifiques. Par exemple, vos sites de New York peuvent uniquement mettre à jour des enregistrements de ventes pour la côte Est des États-Unis (avec des codes postaux locaux inférieurs ou égaux à 49999 selon la fourchette de clés), mais ils peuvent lire tous les enregistrements de ventes.

### Fragmentation par horaire

Concevez votre application de façon à ce que la table puisse être mise à jour uniquement lors de périodes déterminées sur des sites spécifiques. Les périodes doivent être suffisamment séparées pour permettre la réplication des modifications en attente sur le site qui devient la version maître. Pensez à autoriser les changements horaires, comme l'heure d'été, et les différences de zones horaires.

---

## Planification d'une source relationnelle non DB2

Les déclencheurs de capture sont utilisés à la place du programme Capture si vous répliquez à partir de bases de données non DB2. Ces déclencheurs capturent les données modifiées depuis une table source relationnelle non DB2 et les valident dans des tables CCD.

Les déclencheurs de capture affectent les débits et les besoins d'espace journal. Par ailleurs, si vous possédez déjà des déclencheurs dans votre environnement, vous devez éventuellement les fusionner avec les nouveaux déclencheurs de capture. Pour plus d'informations, voir les sections suivantes :

### Débits des transactions pour les déclencheurs de capture

La charge de travail des transactions pour votre système source va en augmentant et la capture des modifications fondée sur des déclencheurs a un impact sur les débits des transactions.

Les déclencheurs de capture augmentent le temps de réponse pour la mise à jour de transactions. L'impact est supérieur pour les transactions qui mettent à jour de façon massive des tables source de l'application à répliquer.

### Impact du journal pour des serveurs source relationnels non DB2

Pour des serveurs source relationnels non DB2, vos applications source ont besoin de plus d'espace de journaux actifs car le volume du journal triple plus ou moins pour des tables source répliquées. Les modifications sont capturées dans les tables source et stockées dans des tables CCD ; les données modifiées sont écrites dans la même portée de validation que les tables source, puis supprimées via un mécanisme de suppression dépendant d'un déclencheur.

Chaque opération INSERT, UPDATE ou DELETE source devient une opération INSERT, UPDATE ou DELETE, ainsi qu'une opération INSERT et une opération DELETE. Le volume du journal augmente encore plus si vous transformez les mises à jour en paires d'opérations DELETE et INSERT.

Si vous ne disposez plus d'espace journal et que le déclencheur Capture ne peut pas insérer un enregistrement dans la table CCD, la transaction tentée par l'utilisateur ou le programme d'application n'aboutit pas.

### Coexistence de déclencheurs existants avec des déclencheurs de capture

La logique des déclencheurs de capture figure dans le script SQL généré par le Centre de réplication lorsque vous enregistrez une source.

Par défaut, des déclencheurs INSERT, UPDATE et DELETE sont créés pour que ces types de modifications (insertion, mise à jour, suppression) puissent être répliqués depuis la table source. Le nom du déclencheur se compose du nom de la table CCD, précédé d'une lettre décrivant le type de déclencheur :I pour INSERT, U pour UPDATE et D pour DELETE. Par exemple, si la table CCD se nomme undjr02.ccd001, le nom du déclencheur DELETE généré est undjr02.dccd001. Vous ne devez pas modifier les noms des déclencheurs générés dans le script.

Si un déclencheur existe déjà dans la table à enregistrer pour réplication et porte le même nom que celui dans le script généré, vous recevez un avertissement au

moment de la génération du script. N'exécutez pas le script généré, car SGBDR risque d'écraser le déclencheur existant. Décidez comment vous souhaitez fusionner les déclencheurs préexistants et les nouveaux, puis créez un script fusionnant votre logique existante et la logique de déclencheur générée par le Centre de répllication.

Si le type de déclencheur à créer existe déjà dans la table à enregistrer pour répllication et que le SGBDR autorise un seul déclencheur de ce type par table, vous devez fusionner la logique avant d'exécuter le script généré.

## Verrous pour des serveurs source Oracle

Toute application qui met à jour le serveur source Oracle doit prendre fin pour que le programme Apply puisse appliquer des données.

Le programme Apply doit verrouiller la table CCD afin de traiter des données et définir le point de synchronisation. Les verrous sur les tables CCD sont uniquement conservés jusqu'à la définition du point de synchronisation, mais pas pendant tout le cycle du programme Apply. Les applications devant mettre à jour la table source attendent que le programme Apply déverrouille la table CCD.

## Modifications nécessaires pour déclencheurs Sybase sur des plateformes little-endian

Lors de la définition des déclencheurs Capture à partir de serveurs Sybase, il est possible que vous deviez modifier les déclencheurs si le système d'exploitation sur lequel est installé Sybase utilise un ordre des octets little-endian.

Les déclencheurs Capture utilisent la fonction @@dbts dans Sybase pour mettre à jour les valeurs synchpoint. La fonction renvoie la valeur de l'horodatage en cours pour la base de données. Cette valeur n'est pas renvoyée correctement si le serveur Sybase s'exécute sur un système d'exploitation qui utilise un ordre des octets little-endian, par exemple Windows NT32. Si Sybase est exécuté sur des plateformes little-endian, les valeurs de la colonne d'horodatage sont affichées en tant que valeurs big-endian, alors que le résultat de la fonction @@dbts reste affichée en tant que valeur little-endian native.

Pour cette raison, vous devez mettre à jour manuellement le script du déclencheur Capture qui est généré par les outils d'administration de répllication si votre serveur Sybase est installé sur ce type de système d'exploitation. Dans le script, modifiez @@dbts comme suit :

```
reverse(substring(@@dbts,1,2)) + 0x0000 + reverse(substring(@@dbts,5,4))
```

---

## Planification de la conversion de pages de codes

Les composants de répllication sont des applications de base de données s'appuyant sur les bases de données DB2 sur divers systèmes d'exploitation pour gérer la traduction de pages de codes de données.

Les composants de répllication fonctionnent avec des données en utilisant les instructions SQL SELECT, INSERT, UPDATE et DELETE.

## Répllication entre des bases de données avec des pages de codes compatibles

Si la configuration de répllication requiert des instructions SQL et des données entre des systèmes avec des pages de codes différentes, les protocoles DB2 tels que

DRDA gèrent la traduction des pages de codes. Par ailleurs, si des données sont transmises entre des bases de données relationnelles DB2 et non DB2, la réplication DB2 s'appuie sur les produits sous-jacents pour gérer la traduction des pages de codes.

Si vous envisagez de répliquer entre des bases de données avec des pages de codes distinctes, consultez le *IBM Information Management Software pour le centre de documentation sur les solutions z/OS* ou le centre de documentation DB2 pour déterminer sur les pages de codes que vous avez sont compatibles. Par exemple, si vous utilisez DB2 for Linux, UNIX, et Windows, reportez-vous à la section sur la conversion des données de type caractère.

Après avoir vérifié que vos bases de données possèdent des pages de codes compatibles, observez si elles les utilisent différemment. Par exemple, si un produit de base de données autorise une page de codes distincte pour chaque colonne dans une table et un autre non, la page de codes doit uniquement être indiquée au niveau de la base de données. Une table comportant plusieurs pages de codes dans le premier produit ne peut alors pas être répliquée sur une base de données dans le second produit. Par conséquent, le mode de gestion des pages de codes par les bases de données affecte la configuration de la réplication, afin que les données soient correctement répliquées entre les diverses bases de données de votre environnement.

## Pages de code pour la réplication SQL

La configuration de la page de codes pour la réplication est définie en même temps que la connectivité de la base de données entre des systèmes. Toutefois, si vous exécutez les programmes Capture ou Apply sous Linux, UNIX ou Windows, vous devrez éventuellement suivre une procédure de configuration.

Sous Linux, UNIX, et Windows, le programme Capture doit s'exécuter dans la même page de codes que la base de données depuis laquelle il capture des données. Si le programme Capture ne s'exécute pas dans la même page de codes, vous devez définir une variable d'environnement ou une variable de registre DB2 nommée DB2CODEPAGE afin que le programme emploie la même page de code que la base de données.

Lorsque vous exécutez le programme Apply sous Linux, UNIX, ou Windows, si toute table source est en UNICODE, le code d'application Apply doit être en UNICODE. Si les données dans la table source sont en ASCII, la page de codes de l'application peut être en ASCII ou en Unicode. Vous pouvez aussi définir la variable DB2CODEPAGE pour le programme Apply.

### Définition de la variable de page de codes

DB2 dérive la page de codes pour une application de l'environnement actif dans lequel celle-ci s'exécute. En général, lorsque la variable DB2CODEPAGE n'est pas définie, elle est dérivée de l'identificateur de langue indiqué par le système d'exploitation. Le plus souvent, cette valeur est correcte pour les programmes Capture ou Apply si vous utilisez la page de codes par défaut au moment de créer votre base de données. Cependant, si vous créez votre base de données avec une page de codes explicite différente de celle par défaut, vous devez définir la variable DB2CODEPAGE. Dans le cas contraire, la traduction de données peut se faire de façon incorrecte. La valeur employée pour la variable DB2CODEPAGE doit être identique à celle indiquée pour votre instruction CREATE DATABASE. Voir le centre d'informations sur DB2 pour plus de détails sur la définition de la variable DB2CODEPAGE.

### Réplication depuis une page de codes

Si vous répliquez des données source avec une page de codes à caractère mono-octet sur une cible avec Unicode UTF-8, certains caractères mono-octet dans la base de données source peuvent être traduits par DB2 en octets doubles ou plus dans la base de données cible. Tous les caractères mono-octet avec une valeur hexadécimale comprise entre 0x80 et 0xff sont traduits dans leur équivalent double octet 1208. Les colonnes cible doivent alors être éventuellement plus grandes que les colonnes source pour éviter que le programme Apply reçoive des erreurs SQL de DB2.

Certains produits de base de données implémentent le support des pages de codes différemment d'autres, ce qui peut avoir une incidence sur votre configuration de réplication. Par exemple, DB2 sur System i permet à une page de codes d'être spécifiée au niveau de la colonne, mais DB2 pour Linux, UNIX, et Windows permet d'indiquer une page de codes uniquement au niveau de la base de données. Par conséquent, si vous avez une table System i avec plusieurs colonnes utilisant différentes pages de codes, ces colonnes ne peuvent être répliquées dans une seule base de données DB2 pour Linux, UNIX et Windows à moins que toutes les pages de codes ne soient compatibles.

### Définition de la variable LANG

Si vous exécutez les programmes Capture et Apply sur un système Linux ou UNIX, vous devez définir la variable d'environnement LANG. Ces programmes utilisent le contenu de cette variable d'environnement pour rechercher la bibliothèque de messages pour votre langue. Par exemple, si la variable d'environnement LANG est en\_US, le programme Capture la recherche dans la bibliothèque de messages en anglais, à l'intérieur du sous-répertoire /sql11b/msg/en\_US de l'instance DB2. Si Capture ne trouve pas sa bibliothèque de messages, tous les messages écrits dans la table IBMSNAP\_CAPTRACE sont ASN0000S.

---

## Planification de la réplication pour DB2 for z/OS

La réplication SQL pour DB2 for z/OS prend en charge les noms de schémas et de tables à hauteur de 128 octets maximum.

Pour bénéficier de la prise en charge de nom long :

- Créez vos tables de contrôle Capture, Apply, et Monitor sous DB2 for z/OS Version 8 ou ultérieure en mode nouvelle fonction.
- Exécutez les serveurs Capture, Apply, et Monitor sous DB2 for z/OS version 8 ou ultérieure en mode nouvelle fonction.

**Restriction :** Si vous souhaitez répliquer vers ou de DB2 for iSeries, vous devez utiliser des noms de schémas de 30 octets au maximum. La réplication sur des plateformes DB2 for z/OS et DB2 for Linux, UNIX et Windows prend en charge des noms de schémas de 128 octets.

---

## Réglages des performances

Le réglage des performances implique celui de votre environnement de réplication pour des performances optimales.

*WebSphere Information Integration Tuning for SQL Replication Performance* décrit comment régler les principaux composants d'un environnement de réplication DB2



pour des performances optimales. Ce document est disponible en ligne sur le site de support de WebSphere Information Integration pour votre produit.



---

## Chapitre 2. Autorisations requises pour la réplication SQL

Pour utiliser les programmes de réplication SQL, vous devez vérifier que les ID utilisateur permettant d'utiliser les programmes de réplication ou les outils de réplication disposent des droits d'accès aux systèmes locaux et distants.

---

### Authentification requise sous Linux, UNIX et Windows

La réplication SQL ne vous oblige pas à utiliser 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 :

- La réplication SQL est un programme d'application de base de données.
- Les fonctions client-serveur DB2 sous-jacentes sont aussi simples pour la réplication SQL que pour les applications de base de données.
- La réplication SQL se connecte aux bases de données DB2 à l'aide des identifiants classiques (ID et mot de passe). Vous pouvez utiliser n'importe quel type d'authentification compatible.
- La réplication SQL ne présente aucune autre exigence relative au type d'authentification.

---

### Autorisations nécessaires pour l'administration

Pour configurer la réplication, vous exécutez le SQL généré pour créer des objets, lier des plans et créer des modules SQL (System i). Les autorisations nécessaires pour ces tâches varient en fonction du système d'exploitation.

Pour administrer la réplication, 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 configurer la réplication. Votre ID utilisateur ne doit pas nécessairement être le même sur tous les systèmes, bien que cela soit préférable pour des raisons de commodité.

z/OS

Linux UNIX Windows

Assurez-vous que les ID utilisateur choisis pour configurer la réplication peuvent exécuter les tâches suivantes :

- Se connecter à tous les serveurs (serveur source, serveur de contrôle Capture, serveur de contrôle Apply, serveur de contrôle Monitor, serveur cible).
- Sélectionner des éléments dans les tables de catalogues sur le serveur source, sur le serveur de contrôle Capture, sur le serveur de contrôle Monitor et sur le serveur cible.
- Créer des tables (y compris des tables de contrôle de réplication), des espaces de table et des vues sur le serveur source, sur le serveur de contrôle Monitor, sur le serveur de contrôle Capture et sur le serveur de contrôle Apply.
- Si vous utilisez des programmes de réplication pour créer de nouvelles tables cible : créer des tables et des espaces de table sur le serveur cible. (Inutile si vous utilisez des tables existantes en tant que cibles).

- Lier des plans ou créer des modules sur chaque base de données DB2 impliquée dans la réplication, y compris le serveur source, le serveur cible, le serveur de contrôle Monitor et le serveur de contrôle Apply.
- Créer des procédures stockées à l'aide d'une bibliothèque partagée et appeler des procédures stockées (Linux, UNIX et Windows uniquement).

Pour les bases de données relationnelles non-DB2, l'ID utilisateur doit être en mesure d'exécuter les actions suivantes :

- Créer des tables.
- Créer des déclencheurs Capture sur des tables source et des tables de contrôle.
- Créer des procédures.
- Créer des pseudonymes sur la base de données fédérée.
- Créer des séquences (pour les bases de données Oracle uniquement).
- Sélectionner des éléments dans des tables de catalogue.

La plupart des administrateurs de réplication ont des privilèges DBADM ou SYSADM. Sous DB2 for z/OS, l'administrateur des rélications doit au minimum être autorisé à sélectionner des éléments dans le catalogue et doit disposer de tous les privilèges nécessaires pour créer des tables avec le schéma ASN et pour créer des tables cible et CD ayant les caractéristiques des tables source, y compris des privilèges de création d'index.

#### System i

Assurez-vous que les ID utilisateur choisis pour configurer la réplication peuvent exécuter les tâches suivantes :

- Se connecter à tous les serveurs (serveur source, serveur de contrôle Capture, serveur de contrôle Apply, serveur de contrôle Monitor, serveur cible).
- Sélectionner des éléments dans les tables de catalogues sur le serveur source, sur le serveur de contrôle Capture, sur le serveur de contrôle Monitor et sur le serveur cible.
- Créer des tables (y compris des tables de contrôle de réplication) et des vues sur le serveur source, sur le serveur de contrôle Monitor, sur le serveur de contrôle Capture et sur le serveur de contrôle Apply.
- Si vous utilisez des programmes de réplication DB2 pour créer de nouvelles tables cible : créer des tables sur le serveur cible. (Inutile si vous utilisez des tables existantes en tant que cibles).
- Lier des plans ou créer des modules sur chaque base de données DB2 impliquée dans la réplication, y compris le serveur source, le serveur cible, le serveur de contrôle Monitor et le serveur de contrôle Apply.

La plupart des administrateurs de réplication ont des privilèges DBADM ou SYSADM.

Utilisez la commande Grant DPR Authority (**GRTDPRAUT**) pour autoriser un utilisateur à enregistrer des sources, à s'abonner à ces sources et à créer des tables de contrôle. Si vous effectuez une réplication entre systèmes System uniquement, vous devez utiliser le même ID utilisateur pour tous les serveurs.

Si la commande Grant DPR Authority (**GRTDPRAUT**) n'est pas installée sur une machine, vous devez utiliser la commande Grant Object Authority (**GRTOBJAUT**).

---

## Autorisations nécessaires pour le programme Capture

L'ID utilisateur qui exécute le programme Capture doit pouvoir accéder au catalogue système DB2, afficher et mettre à jour toutes les tables de contrôle de réplication sur le serveur de contrôle Capture et exécuter les modules du programme Capture.

Vous pouvez utiliser l'ID utilisateur d'administrateur des réplifications pour exécuter le programme Capture, mais cela n'est pas obligatoire.

### z/OS

L'ID utilisateur qui exécute le programme Capture doit être enregistré avec un accès à USS. Cela signifie que cet ID utilisateur doit être défini pour utiliser z/OS UNIX ou OS/390 UNIX (il doit avoir un segment OMVS).

En outre, assurez-vous que la bibliothèque de chargement Capture dispose des droits APF et que l'ID utilisateur qui exécute le programme Capture a les privilèges suivants :

- Accès WRITE sur un répertoire temporaire (le répertoire /tmp ou le répertoire spécifié par la variable d'environnement TMPDIR).
- Privilèges SELECT, UPDATE, INSERT et DELETE pour toutes les tables de réplication sur le serveur de contrôle Capture.
- Privilège SELECT pour le catalogue DB2 (SYSIBM.SYSTABLES, SYSIBM.SYSCOLUMNS. et SYSIBM.SYSPLAN).
- Privilège TRACE.
- Privilèges MONITOR1 et MONITOR2.
- Privilège EXECUTE pour les modules du programme Capture.

Assurez-vous également que l'ID utilisateur dispose du droit d'accès WRITE sur le répertoire capture path (USS) ou sur le qualificatif de haut niveau (z/OS). Pour exécuter le programme Capture dans l'interpréteur de commandes USS, la variable système STEPLIB doit être définie et elle doit inclure la bibliothèque de chargement Capture. Le chemin HFS, /usr/lpp/db2rep1\_10\_01/bin, doit être inclus dans la variable PATH.

### Linux UNIX Windows

Assurez-vous que les ID utilisateur qui exécutent le programme Capture disposent des autorisations et des privilèges suivants :

- Droits DBADM ou SYSADM.
- Privilège WRITE sur le répertoire spécifié par le paramètre **capture\_path**. Le programme Capture crée les fichiers de diagnostic dans ce répertoire.

### Windows

L'ID utilisateur qui exécute le programme Capture doit disposer des droits de création d'objets globaux.

### System i

Utilisez la commande Grant DPR Authority (**GRTDPRAUT**) pour autoriser un utilisateur à exécuter le programme Capture sur un système local. Si vous effectuez une réplication entre systèmes System i uniquement, vous devez utiliser le même ID utilisateur pour tous les serveurs. Si la commande **GRTDPRAUT** n'est pas installée sur une machine, vous devez utiliser la commande Grant Object Authority (**GRTOBJAUT**).

---

## Autorisations nécessaires pour le programme Apply

L'ID utilisateur qui exécute le programme Apply doit pouvoir accéder au catalogue système DB2, afficher et mettre à jour toutes les tables de contrôle de réplication sur le serveur cible et de contrôle Capture et exécuter les modules du programme Apply.

Vous pouvez utiliser différents ID utilisateur sur chaque serveur de votre environnement de réplication. Vous pouvez utiliser l'ID utilisateur d'administrateur des répliqués pour exécuter le programme Apply, mais cela n'est pas obligatoire.

### z/OS

Assurez-vous que les ID utilisateur qui exécutent le programme Apply disposent des autorisations et des privilèges suivants :

- Accès WRITE sur un répertoire temporaire (le répertoire /tmp ou le répertoire spécifié par la variable d'environnement TMPDIR).
- Privilèges SELECT, UPDATE, INSERT et DELETE pour toutes les tables de réplication sur le serveur de contrôle Apply.
- Privilège SELECT pour le catalogue DB2 (SYSIBM.SYSTABLES, SYSIBM.SYSCOLUMNS. et SYSIBM.SYSPLAN).

**Remarque** : l'ID utilisateur qui exécute le programme Apply doit être enregistré avec un accès à USS. Cela signifie que cet ID utilisateur doit être défini pour utiliser z/OS UNIX ou OS/390 UNIX (il doit avoir un segment OMVS). La bibliothèque de chargement doit avoir des droits APF uniquement si le programme Apply doit être enregistré auprès de l'ARM. Pour exécuter le programme Apply dans l'interpréteur de commandes USS, la variable système STEPLIB doit être définie et elle doit inclure la bibliothèque de chargement apply. Le chemin HFS, /usr/lpp/db2repl\_10\_01/bin, doit être inclus dans la variable PATH.

### Linux UNIX Windows

Assurez-vous que les ID utilisateur qui exécutent le programme Apply disposent des autorisations et des privilèges suivants :

- Privilèges WRITE sur le répertoire apply path
- Privilèges d'accès sur les tables source de réplication (y compris les tables CD et CCD associées).
- Privilèges d'accès et de mise à jour sur les tables cible de réplication.
- Privilèges d'accès et de mise à jour sur toutes les tables de contrôle générées par les programmes de réplication et créées sur le serveur de contrôle Capture et le serveur de contrôle Apply.
- Privilèges READ pour tous les fichiers de mots de passe utilisés par le programme Apply.

**Remarque** : si les tables source se trouvent dans un système de gestion de base de données relationnelle non-DB2 : l'ID utilisateur doit avoir des privilèges suffisants sur la base de données fédérée et sur la base de données relationnelle non-DB2, pour accéder aux tables source via des pseudonymes définis dans la base de données fédérée.

### Windows

L'ID utilisateur qui exécute le programme Apply doit disposer des droits de création d'objets globaux.

### System i

Utilisez la commande Grant DPR Authority (**GRTDPRAUT**) pour autoriser un

utilisateur à exécuter le programme Apply sur un système local. Si vous effectuez une réplication entre systèmes System uniquement, vous devez utiliser le même ID utilisateur pour tous les serveurs. Si la commande **GRTDPRAUT** n'est pas installée sur une machine, vous devez utiliser la commande Grant Object Authority (**GRTOBJAUT**).

### Bases de données non-DB2

Si vos tables de contrôle sont des bases de données non-DB2, l'ID utilisateur qui transmet les données modifiées vers une cible relationnelle non-DB2 ou qui en extrait des données, doit disposer de privilèges suffisants sur la base de données fédérée et sur la base de données relationnelle non-DB2.

Pour les cibles relationnelles non-DB2, l'ID utilisateur qui exécute le programme Apply doit avoir les droits nécessaires pour écrire dans les pseudonymes de la base de données fédérée et, via des mappages utilisateur, il doit également disposer des droits nécessaires pour écrire dans la cible non-DB2 elle-même.

Pour les sources relationnelles non-DB2, l'ID qui exécute le programme Apply nécessite les privilèges suivants :

- Privilèges READ et WRITE sur les pseudonymes de la base de données fédérée et, via des mappages utilisateur, privilèges READ et WRITE sur les tables de contrôle Capture.
- Privilège READ sur les pseudonymes de la base de données fédérée et, via des mappages utilisateur, privilège READ sur la table CCD du serveur non-DB2.
- Privilège READ sur les pseudonymes de la base de données fédérée et, via des mappages utilisateur, privilège READ sur la table source du serveur non-DB2.

---

## Autorisations nécessaires pour les déclencheurs Capture sur les bases de données relationnelles non-DB2

Si vous effectuez la réplication à partir d'une base de données non-DB2, les déclencheurs Capture sont utilisés pour identifier les changements par rapport à la source. Les ID utilisateur distants (par exemple, issus des applications utilisateur) qui modifient les tables source distantes doivent disposer des autorisations nécessaires pour effectuer des insertions dans la table CCD.

Dans la plupart des cas, vous n'avez pas besoin d'autorisation explicite pour exécuter des déclencheurs INSERT, UPDATE ou DELETE car, une fois les déclencheurs définis sur une table, leur exécution est transparente pour l'application qui exécute les instructions INSERT, UPDATE or DELETE. Pour les bases de données Informix, les ID d'utilisateurs distants qui exécutent des actions INSERT, UPDATE et DELETE sur la table source enregistrée doivent disposer du privilège EXECUTE PROCEDURE.

Pour les sources Oracle, vous devez octroyer le privilège SELECT pour l'objet de séquence *schéma\_distant.SEQUENCE002*, où *schéma\_distant* correspond au schéma distant dans lequel les tables de contrôle sont créées sous Oracle. L'objet de séquence est créé dans le cadre de la création des tables de contrôle Capture pour une source Oracle et il est utilisé avec les déclencheurs Capture pour remplir la table de modification cohérente des données.

---

## 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	<b>apply_path</b>
Q Apply	<b>apply_path</b>
Q Capture	<b>capture_path</b>
moniteur d'alertes de réplication	<b>monitor_path</b>
Commande asntdiff ou asntrep	<b>DIFF_PATH</b>

- 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 serveurs pour la réplication SQL

Pour pouvoir répliquer des données, vous devez créer et configurer vos serveurs, ainsi que vérifier qu'ils se connectent entre eux.

**z/OS** Pour plus d'informations sur la configuration de serveurs sous z/OS, voir Installation et personnalisation de la réplication pour z/OS.

---

### 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 ultérieure, 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 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 la récupération de 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.

#### Procédure

L'instruction CREATE TABLE ou ALTER TABLE permet de définir DATA CAPTURE CHANGES sur les tables source de réplication. Pour les tables de catalogue système DB2 for z/OS, cette opération est automatique lorsque vous créez des tables de contrôle à l'aide du Centre de réplication version 10 ou du programme ASNCLP. Vous pouvez également utiliser les instructions suivantes :

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

---

## Besoins de connectivité pour la réplication SQL

Tout poste de travail exécutant le programme Apply, le Centre de réplication ou les commandes de réplication doit pouvoir se connecter aux bases de données du serveur source, du serveur de contrôle de Capture, du serveur de contrôle Apply et du serveur cible.

Si vous utilisez le moniteur d'alertes de réplication, le poste de travail sur lequel il s'exécute doit pouvoir se connecter au serveur de contrôle de Monitor et à tout serveur qu'il contrôle. Pour utiliser le Centre de réplication en vue de configurer le contrôle, vérifiez que celui-ci peut se connecter au serveur de contrôle de Monitor.

Si votre conception de réplication implique le transfert de données sur un serveur autre que la base de données source, vous devez étudier avec soin les communications entre les serveurs. Veillez à limiter les couches d'émulation, les ponts vers le réseau local et les liaisons de routeur, sachant que tous peuvent affecter les performances de réplication.

Si les bases de données sont connectées à un réseau, la connectivité varie en fonction des systèmes d'exploitation connectés.

Les rubriques suivantes offrent des détails sur les besoins de connectivité.

## Connexion aux serveurs System i à partir de Windows

Vous pouvez gérer la réplication sur des serveurs System i en vous connectant à partir d'un poste de travail Windows.

### Avant de commencer

- Vous devez avoir DB2 ou DB2 Connect installé sur votre poste de travail.
- Vous devez avoir un protocole TCP/IP installé sur votre poste de travail.

### Procédure

Pour vous connecter à un serveur System i à partir d'un poste de travail DB2 pour Windows :

1. Connectez-vous au serveur System i et localisez la base de données relationnelle :
  - a. Connectez-vous au serveur System i auquel vous voulez vous connecter.
  - b. Soumettez une commande **dsprdbdir**, puis indiquez `local` pour `*LOCAL`.
  - c. Localisez le nom de la base de données relationnelle dans la sortie. Par exemple, dans la sortie suivante, la base de données est appelée DB2400E :

```
MYDBOS2          9.112.14.67  
RCHASDPD         RCHASDPD  
DB2400E          *LOCAL  
RCHASLJN         RCHASLJN
```

2. Cataloguez la base de données System i dans DB2 pour Windows :
  - a. A partir d'une invite de commande Windows, saisissez **db2cmd**. La fenêtre de la commande CLP DB2 s'ouvre.
  - b. Dans la fenêtre de la commande, saisissez les trois commandes suivantes en respectant l'ordre exact :

```
db2 catalog tcpip node server_name remote server_name server 446 system
server_name ostype OS400
```

```
db2 catalog dcs database rdb_name AS rdb_name
```

```
db2 catalog database rdb_name AS rdb_name at node server_name
authentication dcs
```

Où *server\_name* est le nom d'hôte TCP/IP du système System i, et *rdb\_name* est le nom de la base de données relationnelle System i que vous avez trouvée à l'étape 1.

3. Pour quitter le programme command, émettez la commande suivante :  
db2 terminate
4. Veillez à ce que le profil utilisateur System i que vous utiliserez pour vous connecter à votre système System i utilise CCSID37 :
  - a. Connectez-vous au système System i.
  - b. Saisissez la commande suivante, dans laquelle *user* est le profil utilisateur :  
CHGUSRPRF USRPRF (*user*) CCSID(37)
  - c. Veillez à ce que le serveur DDM démarre sur le type de système System i :  
STRTCPSVR SERVER(\*DDM)
5. Veillez à ce que DB2 pour Windows et DB2 pour System i soient connectés :  
db2 connect to *rdb\_name* user *user\_name* using *password*

## Connexion à des serveurs relationnels non DB2

Pour répliquer des données de ou vers un serveur relationnel non DB2, vous devez avoir accès à ce serveur et pouvoir vous y connecter.

Avant de tenter la réplification depuis des serveurs source relationnels non DB2, vous devez configurer votre serveur fédéré et votre base de données. Il y a trois étapes principales à suivre :

1. Définissez un encapsuleur pour que la base de données DB2 puisse accéder à d'autres bases de données relationnelles non DB2.
2. Définissez une base de données relationnelle non DB2 avec un mappage de serveur.

**Restriction :** Le Centre de réplification et le programme de ligne de commande ASNCLP ne prennent pas en charge la création des tables de contrôle ni des tables cibles dans les bases de données Oracle si la validation en deux phases du mappage de serveur est activée.

3. Si la combinaison ID utilisateur/mot de passe employée pour la connexion à la base de données DB2 est différente de celle utilisée pour accéder à la base de données relationnelle non DB2, vous devez créer un mappage utilisateur.

Pour plus de détails sur la configuration d'un environnement fédéré, voir le centre d'informations sur DB2 ou *WebSphere Information Integration Federated Systems Guide*.

---

## Création de tables de contrôle pour la réplification SQL

Les programmes de réplification utilisent des tables de contrôle pour stocker des informations sur les tables enregistrées, les ensembles d'abonnements, les paramètres facultatifs et les préférences utilisateur. Vous créez des tables de contrôle avant de définir vos sources et vos cibles pour la réplification.

## Création de tables de contrôle pour la réplication SQL

Vous pouvez utiliser le Centre de réplication ou le programme de ligne de commande ASNCLP pour créer des tables de contrôle pour les programmes Capture et Apply.

### Restrictions

- Le Centre de réplication ou ASNCLP doivent pouvoir se connecter au serveur sur lequel vous voulez créer les tables de contrôle.
- Dans un environnement avec plusieurs partitions de base de données, tous les espaces table utilisés par les tables de contrôle doivent figurer sur la partition de catalogue. Si vous utilisez un espace table existant, celui-ci ne doit pas être partitionné et doit figurer dans la partition du catalogue.

### A propos de cette tâche

Si vous ne personnalisez pas le mode de création des tables de contrôle, deux espaces table sont créés, l'un pour la table UOW, l'autre pour les autres tables de contrôle. Si vous ne souhaitez pas utiliser les espaces table par défaut, vous pouvez indiquer des espaces table existants, en créer ou employer l'espace table DB2 en cours.

Si Capture est démarré dans un environnement avec plusieurs partitions de base de données, il crée une autre table de contrôle (IBMSNAP\_PARTITIONINFO) dans le même espace table que la table IBMSNAP\_RESTART.

Les deux outils d'administration de réplication vous permettent de créer un profil et d'identifier les valeurs par défaut à utiliser lorsque vous créez des tables de contrôle pour un système d'exploitation ou un environnement déterminé. Une fois les profils définis pour ces tables de contrôle, il est inutile de les définir pour chaque ensemble de tables de contrôle créé. Vous pouvez remplacer les valeurs par défaut lorsque vous créez les tables de contrôle. Vous pouvez aussi modifier à tout moment le profil, mais les modifications ne concerneront que les tables de contrôle créées après changement du profil.

### Procédure

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

Méthode	Description
<b>Programme de ligne de commande ASNCLP</b>	Utilisez la commande CREATE CONTROL TABLES FOR pour créer un ensemble de tables de contrôle Capture ou Apply. Par exemple, les commandes suivantes définissent l'environnement et créent des tables de contrôle Capture : <pre>SET SERVER CAPTURE TO DB SAMPLE SET OUTPUT CAPTURE SCRIPT "capctrl.sql"; SET LOG "capctrl.err"; SET RUN SCRIPT LATER; CREATE CONTROL TABLES FOR CAPTURE SERVER IN UW UOW TSUOW100 OTHERS TSASN100;</pre>

Méthode	Description
Centre de réplication	<p>Utilisez les fenêtres Création de tables de contrôle de Capture - Rapide ou Création de tables de contrôle d'Apply - Rapide. Pour ouvrir ces fenêtres, cliquez avec le bouton droit sur un dossier Serveurs de contrôle de Capture ou Serveurs de contrôle d'Apply dans l'arborescence des objets, puis sur l'une des options suivantes :</p> <ul style="list-style-type: none"> <li>• Création de tables de contrôle de Capture</li> <li>• Création de tables de contrôle de Capture &gt; Rapide</li> <li>• Création de tables de contrôle d'Apply</li> <li>• Création de tables de contrôle d'Apply &gt; Rapide</li> </ul>

## Création de tables de contrôle (System i)

Les tables de contrôle de réplication sont automatiquement créées lorsque vous installez DB2 DataPropagator pour System i. Vous pouvez également utiliser une commande pour créer des tables de contrôle.

### A propos de cette tâche

Pendant l'installation, si elles n'existent pas déjà, les tables de contrôle sont créées dans le schéma par défaut DataPropagator (ASN). Si vos tables de contrôle sont accidentellement supprimées ou corrompues, vous pouvez créer des ensembles de tables de contrôle supplémentaires. Pour Capture, vous pouvez créer le nouvel ensemble de tables de contrôle avec un schéma différent. Vous pouvez créer un maximum de 25 schémas.

Pour un système de fichiers défini par l'utilisateur, vous pouvez créer les tables de contrôle de réplication dans le groupe ASP (Auxiliary Storage Pool) ou IASP (Independent Auxiliary Storage Pool) de base, mais pas dans les deux. SI vous créez les tables de contrôle dans un groupe IASP, vous devez d'abord supprimer toutes les tables de contrôle Capture et Apply de la base ASP. Emettez la commande **SETASPGRP** pour le groupe ASP qui contient la bibliothèque ASN (ou toute autre bibliothèque pour un schéma Capture) avant de démarrer les programmes Capture ou Apply.

### Procédure

Pour créer des tables de contrôle sur System i, utilisez la commande Create DPR Tables (**CRTDPRTBL**).

**Restriction :** Utilisez uniquement la commande **CRTDPRTBL** pour créer des tables de contrôle sur System i. Ni le programme de ligne de commande ASNCLP ni le Centre de réplication ne prennent en charge la création des tables de contrôle pour System i.

## Création de tables de contrôle pour des sources relationnelles non DB2

Pour utiliser une base de données non DB2 comme Informix en tant que source de réplication, vous pouvez recourir au Centre de réplication ou au programme de ligne de commande ASNCLP afin de créer des tables de contrôle.

### A propos de cette tâche

Pour ces types de sources, le Centre de réplication crée les tables de contrôle Capture ci-après dans la base de données relationnelle non DB2 :

- IBMSNAP\_PRUNCNTL
- IBMSNAP\_PRUNE\_SET
- IBMSNAP\_REG\_SYNCH
- IBMSNAP\_REGISTER
- IBMSNAP\_SEQTABLE sur Informix uniquement
- IBMSNAP\_SIGNAL

Des pseudonymes sont créés dans une base de données fédérée, sauf pour IBMSNAP\_SEQTABLE. (Cette table est uniquement utilisée par les déclencheurs Informix. Le programme Apply ne l'emploie pas.) Les déclencheurs sont créés automatiquement dans les tables IBMSNAP\_SIGNAL et IBMSNAP\_REG\_SYNCH.

**Important :** ne supprimez pas et ne modifiez pas les déclencheurs créés dans les tables IBMSNAP\_SIGNAL et IBMSNAP\_REG\_SYNCH.

## Création de plusieurs ensembles de tables de contrôle Capture

Pour utiliser plusieurs programmes Capture sur un serveur, vous devez créer plusieurs ensembles de tables de contrôle Capture et vérifier que chacun possède un schéma de Capture unique.

### A propos de cette tâche

Ce schéma identifie le programme Capture utilisant un ensemble de tables. Plusieurs schémas de Capture vous permettent d'exécuter à la fois divers programmes Capture.

Vous pouvez exécuter plusieurs programmes Capture dans les cas suivants :

- Pour optimiser les performances en traitant les tables de faible temps d'attente différemment des autres. Si vous possédez des tables de faible temps d'attente, vous pouvez les répliquer avec leur propre programme Capture. De cette façon, vous pouvez leur attribuer une priorité d'exécution. De plus, vous pouvez définir les paramètres du programme Capture, comme l'intervalle de suppression et celui de contrôle, pour respecter le faible temps d'attente de ces tables.
- Pour un meilleur rendement potentiel de Capture. Il peut s'agir d'un avantage de poids dans un environnement source comptant plusieurs unités centrales. Pour un rendement accru, un temps système supplémentaire doit être associé à plusieurs programmes de lecture du journal.

Pour une réplication à partir de plusieurs bases de données source non DB2 dans la même base de données fédérée, vous devez créer des ensembles de tables de contrôle Capture, chacun avec son propre schéma. Si vous préférez, vous pouvez utiliser des bases de données fédérées distinctes, auquel cas les tables de contrôle Capture sur chaque serveur peuvent employer le schéma ASN par défaut.

**z/OS** Vous pouvez utiliser plusieurs schémas de Capture pour utiliser séparément des algorithmes de codage UNICODE et EBCDIC ou pour exécuter plusieurs instances du programme Capture sur un sous-système.



Utilisez la commande de création de tables DPR (**CRTPRTBL**) pour créer un autre ensemble de tables de contrôle Capture, en utilisant le paramètre **CAPCTLLIB** pour spécifier le nom du schéma.

## Création de tables de contrôle dans une base de données à partitions multiples

Lorsque vous créez des tables de contrôle Capture dans une base de données à plusieurs partitions, vous devez les placer dans un espace table à partition unique sur la partition de catalogue.

Vous devez d'abord créer un espace table à partition unique, puis indiquer cet espace table lorsque vous utilisez le Centre de réplication ou le programme de ligne de commande **ASNCLP** pour créer les tables de contrôle.

Si vous démarrez le programme Capture pour la première fois et sélectionnez le mode de démarrage **WARMSI**, la table **IBMSNAP\_PARTITIONINFO** n'existe pas. Le programme Capture crée cette table et un index à entrées uniques dans l'espace table où la table **IBMSNAP\_RESTART** figure. Une fois la table **IBMSNAP\_PARTITIONINFO** créée, le programme Capture insère une ligne pour chaque partition de base de données.

S'il ne s'agit pas du premier démarrage du programme Capture et que vous sélectionnez l'un des modes de démarrage à chaud, la table **IBMSNAP\_PARTITIONINFO** existe déjà. Dans le Centre de réplication, si vous avez coché la case **Une ou plusieurs partitions ont été ajoutées depuis la dernière exécution de Capture**, le programme Capture insère une ligne dans la table **IBMSNAP\_PARTITIONINFO** pour chaque partition de base de données ajoutée depuis sa dernière exécution.

---

## Configuration des programmes de réplication

En vue de la réplication, vous devez configurer le programme Capture, le programme Apply et d'autres programmes de réplication pour les serveurs de votre environnement.

Les rubriques suivantes décrivent la configuration requise pour les programmes de réplication.

### Configuration des programmes de réplication (Linux, UNIX, Windows)

Pour configurer les programmes de réplication nécessaires en vue de définir des variables d'environnement, préparez la base de données pour le programme Capture et lier éventuellement des packages.

#### Définition de variables d'environnement pour les programmes de réplication (Linux, UNIX, Windows)

Vous devez définir des variables d'environnement avant de démarrer et d'arrêter le programme Capture, le programme Apply ou le programme de moniteur d'alertes de réplication, et avant d'utiliser le Centre de réplication ou des commandes système de réplication.

#### Procédure

Pour définir les variables d'environnement :

1. Définissez la variable d'environnement pour le nom d'instance DB2 (DB2INSTANCE) comme suit :

Systeme d'exploitation	Commande
Linux UNIX	export DB2INSTANCE= <i>nom_instance_db2</i>
Windows	SET DB2INSTANCE= <i>nom_instance_db2</i>

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. **Remarque :** Capture doit s'exécuter dans la même page de codes que la base de données pour laquelle il capture des données. DB2 dérive la page de codes de Capture de l'environnement actif où le programme s'exécute. Si la variable DB2CODEPAGE n'est pas définie, DB2 dérive la valeur de la page de codes du système d'exploitation. Dans ce cas, la valeur est correcte pour Capture si vous avez employé la page de codes par défaut au moment de créer la base de données.
3. Facultatif : Définissez la variable d'environnement DB2DBDFT sur le serveur source.
4. **Linux UNIX** Vérifiez que les variables du chemin de la bibliothèque et du chemin d'exécutable propres au système incluent le répertoire dans lequel les bibliothèques et les exécutables requis sont installés.

### Préparation de la base de données DB2 pour exécuter le programme Capture (Linux, UNIX, Windows)

Pour préparer la base de données DB2 afin d'exécuter le programme Capture, vous activez la consignation d'archivage. Vous pouvez également définir d'autres paramètres de configuration de la base de données.

#### Procédure

Pour préparer la base de données DB2 en vue d'exécuter le programme Capture :

1. 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 modifiant la valeur du paramètre de configuration de base de données LOGARCHMETH1 sur une valeur autre que OFF.  
Pour les environnements avec plusieurs partitions de base de données, chaque partition doit être configurée pour permettre la récupération aval si le programme Capture en capture des modifications.
2. Vous devez éventuellement augmenter les valeurs de configuration en fonction des besoins d'installation.
  - Pour les transactions avec un grand nombre de lignes ou des lignes très volumineuses, il est conseillé d'augmenter la valeur du paramètre **memory\_limit** de Capture.
  - Les valeurs suivantes de configuration de la base de données sont appropriées pour de nombreux scénarios de postes de travail de grande taille : APPLHEAPSZ 1000, LOGFILSIZ 4000, LOGPRIMARY 8, LOGSECOND 40, DBHEAP 1000, LOGBUFSZ 16, MAXAPPLS 200.

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

Le programme Capture est automatiquement lié sur Linux, UNIX et Windows au cours de l'exécution. Vous pouvez lier manuellement des packages si vous souhaitez indiquer des options de liaison, planifier une liaison ou vérifier que tous les processus ont abouti.

### Procédure

Pour lier les packages du programme Capture :

1. Connectez-vous à la base de données du serveur de contrôle Capture en entrant la commande suivante :

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

où *base de données* correspond à la base de données du serveur de contrôle Capture.

2. Allez au répertoire dans lequel figurent les fichiers de liens du programme Capture.

#### Linux UNIX

```
répertoire personnel db2/sql11ib/bnd
```

où *répertoire personnel db2* correspond au répertoire personnel de l'instance DB2.

#### Windows

```
unité:\...\sql11ib\bnd
```

où *unité*: correspond à l'unité où DB2 est installé.

3. Créez et liez le package du programme Capture à la base de donnée du serveur source en entrant la commande suivante :

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

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

Ces commandes créent des packages, dont les noms figurent dans le fichier `capture.lst`.

## Facultatif : liaison des packages du programme Apply (Linux, UNIX, Windows)

Le programme Apply est automatiquement lié sur Linux, UNIX et Windows au cours de l'exécution. Vous pouvez lier manuellement des packages si vous souhaitez indiquer des options de liaison, planifier une liaison ou vérifier que tous les processus ont abouti.

### Procédure

Pour lier les packages du programme Apply :

1. Allez au répertoire dans lequel figurent les fichiers de liens du programme Apply.

#### Linux UNIX

```
répertoire personnel db2/sql11ib/bnd
```

où *répertoire personnel db2* correspond au répertoire personnel de l'instance DB2.

*unité*: \...\sql1lib\bnd

où *unité*: correspond à l'unité où DB2 est installé.

2. Pour chaque serveur source, serveur cible, serveur de contrôle de Capture et serveur de contrôle Apply, procédez comme suit :

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

où *base de données* correspond au serveur source, au serveur cible, au serveur de contrôle de Capture ou au serveur de contrôle Apply. 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**. Par exemple :

```
db2 connect to base de données user
IDutilisateur using mot de
pas
```

3. Créez et liez le package du programme Apply à la base de données en entrant les commandes suivantes :

```
db2 bind @applycs.lst isolation cs blocking all grant public
db2 bind @applyur.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 *applycs.lst* et *applyur.lst*.

## Création de modules SQL à utiliser avec des systèmes distants (System i)

Vous devez créer des modules utilisant dans certains cas la commande **CRTSQLPKG**, sur System i.

### A propos de cette tâche

Utilisez cette commande pour créer des modules dans les cas suivants :

- Lors de l'utilisation de la journalisation distante. Exécutez la commande **CRTSQLPKG** sur le système lorsque le programme Capture s'exécute et pointe vers le système sur lequel se trouve la table source.
- Avant d'utiliser la commande **ADDDPRSUB** ou **ADDDPRSUBM** pour ajouter un ensemble d'abonnements ou un membre d'un ensemble d'abonnements. Exécutez la commande **CRTSQLPKG** sur le serveur cible et utilisez les directives suivantes :
  - Si la table source est située sur un autre ordinateur, pointez vers le système sur lequel est situé la table source.
  - Si le serveur de contrôle Apply se trouve sur une autre machine, pointez vers le serveur de contrôle Apply.

Les modules SQL permettront aux programmes de réplication de fonctionner dans un environnement de réplication réparti, que cet environnement soit celui dans lequel vous répliquez entre les systèmes System i ou entre un système System i et d'autres systèmes d'exploitation (tels que Linux, UNIX, ou Windows).

Pour plus d'informations sur l'utilisation de la commande **CRTSQLPKG**, voir la Programmation SQL DB2 pour i5/OS.

Les modules sont créés à l'aide du qualificatif ASN. Sur System i ils sont créés dans la bibliothèque ASN. Sur les autres systèmes d'exploitation, ils sont créés dans le schéma ASN.

### **Création de modules SQL pour le programme Apply (System i)**

Vous devez créer des modules SQL pour le programme Apply sur System i afin qu'il puisse interagir avec tous les serveurs distants auxquels il doit se connecter.

#### **Procédure**

Pour créer des modules SQL pour le programme Apply :

Exécutez cette commande sur le système sur lequel Apply est en cours d'exécution, afin de lui permettre de se connecter à un système distant :

```
CRTSQLPKG PGM(QDP4/QZSNAPV2) RDB(remote_system)
```

Où *remote\_system* est le nom d'entrée de base de données relationnelle pour le système distant auquel le programme Apply doit se connecter.

### **Création de modules SQL pour Replication Analyzer (System i)**

Vous devez créer des modules SQL pour le Replication Analyzer sur System i afin qu'il puisse interagir avec tous les serveurs que vous analysez tels que le serveur de contrôle Capture sur le serveur cible.

#### **Procédure**

Pour créer des modules SQL pour le Replication Analyzer :

Exécutez la commande suivante sur le système sur lequel s'exécutera Replication Analyzer :

```
CRTSQLPKG PGM(QDP4/QZSNANZR) RDB(remote_system)
```

Où *remote\_system* est le nom du système que vous analysez.

### **Concession de privilèges sur les modules SQL (System i)**

Après avoir créé des modules SQL sur System i, vous devez accorder des privilèges \*EXECUTE à tous les utilisateurs qui s'abonnent aux fichiers enregistrés sur la base de données source.

#### **Procédure**

Pour accorder des privilèges pour les modules SQL :

Connectez-vous sur le système System i sur lequel réside la base de données source et utilisez l'une des méthodes suivantes :

Méthode	Description
Commande <b>GRTOBJAUT</b>	Utilisez la commande Grant Object Authority ( <b>GRTOBJAUT</b> ) par exemple : <pre>GRTOBJAUT OBJ(ASN/package_name) OBJTYPE(*SQLPKG) USER(subscriber_name) AUT(*OBJOPR *EXECUTE)</pre>

Méthode	Description
<b>instruction GRANT SQL</b>	Utilisez SQL pour vous connecter à la base de données source et exécuter l'instruction GRANT SQL : CONNECT TO <i>data_server_RDB_name</i> GRANT EXECUTE ON PACKAGE <i>ASN/package_name</i> TO <i>subscriber_name</i>
<b>Commande GRTDPRAUT</b>	Utilisez la commande <b>GRTDPRAUT</b> , si elle est installée sur le système local.

## Paramétrage des programmes de réplication (z/OS)

Lorsque vous installez SQL sur z/OS, vous devez paramétrer et personnaliser les programmes de réplication.

Reportez-vous aux instructions de la section Installation et personnalisation de la réplication pour z/OS.

## Capture de plusieurs partitions de base de données

Si vous répliquez des données sur le serveur DB2 Enterprise Server Edition, vous pouvez capturer des modifications des tables source réparties entre plusieurs partitions de base de données.

Le programme Capture conserve une liste des partitions de base de données appartenant au groupe de partitions dans la table IBMSNAP\_PARTITIONINFO. Cette table est créée par le programme Capture la première fois que celui-ci démarre et détecte plusieurs partitions de base de données dans son groupe de partitions.

Chaque fois que le programme Capture fait un démarrage à chaud, il lit la liste des partitions de base de données du groupe de partitions où figurent les tables de contrôle. Le programme Capture compare le nombre de partitions de base de données connues par DB2 à celui indiqué dans la table IBMSNAP\_PARTITIONINFO. Le nombre de partitions dans la table IBMSNAP\_PARTITIONINFO doit correspondre à celui connu par DB2 afin que le programme Capture s'exécute.

Si vous avez ajouté une ou plusieurs partitions de base de données depuis la dernière exécution du programme Capture, vous devez lui indiquer ces nouvelles partitions. Vous pouvez pour cela utiliser le Centre de réplication en cochant la case **Une ou plusieurs partitions ont été ajoutées depuis la dernière exécution de Capture** au moment de définir le paramètre **startmode** pour l'un des modes de démarrage à chaud dans la fenêtre Démarrer Capture.

## Réplication de tables partitionnées : version 9.7 avec groupe de correctifs 1 ou précédent (Linux, UNIX, Windows)

La réplication SQL prend en charge les tables DB2 qui sont partitionnées par intervalle (en utilisant la clause PARTITION BY de l'instruction CREATE TABLE). Ces tables sont parfois connues sous le nom de tables partitionnées par intervalle.

Il existe des exigences concernant la version et le groupe de correctifs du programme Capture lorsqu'une table source est partitionnée par intervalle. La présente rubrique couvre les exigences et la prise en charge de la réplication des tables partitionnées lorsque vos programmes de réplication sont à la version 9.7 groupe de correctifs 1 ou une version antérieure. Si seules les tables cible (et non

les tables source) sont partitionnées par intervalle, il n'existe alors aucune exigence de version ou de groupe de correctifs spécifique à ces tables.

Si vous souhaitez capturer des données modifiées pour des tables partitionnées par intervalle, vous devez disposer du programme Capture version 9.7 ou ultérieure. Il peut capturer les modifications depuis les tables partitionnées par intervalle sur DB2 version 9.1, 9.5 ou 9.7. Cependant, des restrictions existent pour les tables partitionnées par intervalle antérieures à la version 9.7. Ces restrictions sont également traitées dans la présente 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 partitions ou intervalles de données, en fonction des valeurs contenues dans une ou plusieurs colonnes clés de partitionnement de la table.

La réplication SQL traite toutes les partitions de données d'une table source comme une seule table. Par exemple, lorsque vous enregistrez 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.

Vous pouvez apporter des modifications à une table partitionnée, notamment ajouter une partition de données, joindre une partition de données ou déconnecter une partition de données. 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 SQL traite ces opérations ALTER différemment :

**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 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. Capture commence automatiquement à répliquer les modifications sur la partition de données.

**Version 9.5 ou 9.1**

Capture ne reconnaît pas l'ajout de la partition de données jusqu'à ce que le programme soit réinitialisé ou arrêté et redémarré. Ajoutez la partition de données sur la source et la cible avant de redémarrer Capture. Ne modifiez pas les données dans la partition source jusqu'à ce que 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 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, vous devez les charger manuellement.

**Remarque :** Si le programme Capture est arrêté lorsqu'une partition de données est jointe, les lignes insérées, mises à jour ou supprimées sur la table avant qu'elle ne soit jointe sont répliquées. Si Capture est en cours de fonctionnement lorsque la partition de données est jointe, ces lignes ne

sont pas répliquées.

Pour garantir un comportement cohérent, avant de joindre une table comme nouvelle partition de données, définissez la clause DATA CAPTURE CHANGES de la table sur OFF si vous avez besoin d'y apporter des modifications. 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. Le programme Capture traite également ces mises à jour en tant que suppressions et insertions dans la table de modification des données (similaire au comportement de CHG\_UPD\_TO\_DEL\_INS=Y dans la table IBMSNAP\_REGISTER).

## Réplication de tables partitionnées DB2 : version 9.7 avec groupe de correctifs 2 ou ultérieur (Linux, UNIX, Windows)

La réplication SQL prend en charge les tables DB2 qui sont partitionnées par intervalle (en utilisant la clause PARTITION BY de l'instruction CREATE TABLE). Ces tables sont parfois connues sous le nom de tables partitionnées par intervalle.

Il existe des exigences concernant la version et le groupe de correctifs du programme Capture lorsqu'une table source est partitionnée par intervalle. La présente rubrique couvre les exigences et la prise en charge de la réplication des tables partitionnées lorsque vos programmes de réplication sont à la version 9.7 groupe de correctifs 2 ou une version ultérieure. Si seules les tables cible (et non les tables source) sont partitionnées par intervalle, il n'existe alors aucune exigence de version ou de groupe de correctifs spécifique à ces tables.

Si vous souhaitez capturer des données modifiées pour des tables partitionnées par intervalle, vous devez disposer du programme Capture version 9.7 ou ultérieure. Il peut capturer les modifications depuis les tables partitionnées par intervalle sur DB2 version 9.1, 9.5 ou 9.7. Cependant, des restrictions existent pour les tables partitionnées par intervalle antérieures à la version 9.7. Ces restrictions sont également traitées dans la présente rubrique.

**Important :** Si la version de vos programmes de réplication est 9.7 avec groupe de correctifs 2 ou ultérieur, et que vous prévoyez de répliquer des tables partitionnées par intervalle, vous devez exécuter le script de migration version 9.7 avec groupe de correctifs 2, `asncap1uwv97fp2.sql`. Le script ajoute une nouvelle table de contrôle, `IBMQREP_PART_HIST`, pour aider les programmes de réplication à gérer les modifications des partitions de données tels que l'ajout, la connexion ou la



déconnexion. Le script se trouve dans le répertoire `samples/rep1/mig97/sql/`. Le programme Capture n'utilise pas la table `IBMQREP_PART_HIST` pour les tables source partitionnées sur DB2 version 9.5 ou 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 partitions ou intervalles de données, en fonction des valeurs contenues dans une ou plusieurs colonnes clés de partitionnement de la table.

La réplication SQL traite toutes les partitions de données d'une table source comme une seule table. Par exemple, lorsque vous enregistrez 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.

Vous pouvez apporter des modifications à une table partitionnée, notamment ajouter une partition de données, joindre une partition de données ou déconnecter une partition de données. 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 SQL traite ces opérations ALTER différemment :

**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 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. Capture commence automatiquement à répliquer les modifications sur la partition de données.

**Version 9.5 ou 9.1**

Capture ne reconnaît pas l'ajout de la partition de données jusqu'à ce que le programme soit réinitialisé ou arrêté et redémarré. Ajoutez la partition de données sur la source et la cible avant de redémarrer Capture. Ne modifiez pas les données dans la partition source jusqu'à ce que 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 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, vous devez les charger manuellement.

**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. Le programme Capture traite également ces mises à jour en tant que suppressions et

insertions dans la table de modification des données (similaire au comportement de CHG\_UPD\_TO\_DEL\_INS=Y dans la table IBMSNAP\_REGISTER).

## Exécution de DB2 Query Patroller dans un environnement de réplication SQL

Si vous configurez la réplication dans un environnement dans lequel DB2 Query Patroller s'exécute également, vous devez suivre une procédure spéciale afin de ne pas compromettre les activités de réplication.

### A propos de cette tâche

Query Patroller contrôle les coûts liés aux requêtes dynamiques émises pour une base de données. Si vous exécutez le client Query Patroller et que le coût d'une requête dépasse le seuil défini par l'administrateur de base de données, la requête est interrompue et un message est généré.

Les composants de la réplication peuvent émettre un grand nombre de requêtes dynamiques. Si Query Patroller est activé et que l'emplacement d'installation du client correspond à l'emplacement d'exécution des composants de la réplication, les situations suivantes peuvent se produire :

- Des messages sont générés sur le système client lorsqu'une requête dynamique de réplication dépasse le seuil défini.
- Si Query Patroller identifie une erreur, il peut envoyer un code SQLCODE différent de zéro aux composants de réplication. La valeur de la plupart des codes SQLCODE est comprise entre SQL29000N et SQL29999N.

### Procédure

Pour éviter ce type de situations, procédez comme suit :

- Exécutez les composants de réplication à partir d'un client DB2 dont le client Query Patroller n'est pas activé.
- Désactivez Query Patroller. Désactivez-le par exemple pour une base de données donnée en définissant le paramètre DYN\_QUERY\_MGMT sur 0 (DISABLE). DYN\_QUERY\_MGMT est un paramètre de configuration de base de données qui détermine si Query Patroller est activé pour une base de données en particulier.

## Configuration des journaux (System i)

DB2 DataPropagator pour System i utilise les informations qu'il reçoit des journaux concernant les modifications des données afin de renseigner les tables de données de modification et tables des unités d'oeuvre pour la réplication.

DB2 DataPropagator pour System i s'exécute sous contrôle de validation pour la plupart des opérations et exige donc des fonctions journal dans les tables de contrôle. (Le journal QSQRN est créé lorsque la commande **CRTDPRTBL** crée une collection.)

Les administrateurs doivent s'assurer que les bibliothèques contenant la table source, une table de données de modification et une table cible contiennent des journaux. Ils doivent également veiller à ce que toutes les tables source soient correctement journalisées.

Avant d'enregistrer une table pour la réplication sur System i, la table doit être journalisée pour les images avant et les images après.

Les rubriques suivantes décrivent la configuration de journal requise pour la réplication.

## Configuration des journaux pour les tables source (System i)

Pour configurer la journalisation d'une table source, vous créez un récepteur de journal, ainsi qu'un journal, puis vous lancez la journalisation.

### Avant de commencer

Vous devez bénéficier des droits d'accès pour créer des journaux et récepteurs de journal pour les tables source à définir.

### Restrictions

Pour les tables source, utilisez un journal différent de celui créé par DB2 DataPropagator pour System i dans la bibliothèque du schéma ASN (ou autre schéma de Capture).

### Procédure

Pour créer un journal de table source :

1. Créez un récepteur de journal dans une bibliothèque de votre choix à l'aide de la commande Create Journal Receiver (**CRTJRNRCV**). Placez le récepteur de journal dans une bibliothèque qui est régulièrement enregistrée. Sélectionnez un nom de récepteur de journal pouvant être utilisé pour créer une convention de dénomination pour les futurs récepteurs de journal, tel que RCV0001. Vous pouvez utiliser l'option \*GEN pour poursuivre la convention de dénomination lorsque vous modifiez vos récepteurs de journal. Ce type de convention de dénomination est également utile si vous choisissez de laisser le système gérer la modification de vos récepteurs de journal. L'exemple suivant utilise un bibliothèque nommée JRNLIB pour les récepteurs de journal.

```
CRTJRNRCV  JRNRCV(JRNLIB/RCV0001)
           THRESHOLD(100000)
           TEXT('DataPropagator Journal Receiver')
```

2. Créez le journal en utilisant la commande Create Journal (**CRTJRN**), comme dans l'exemple suivant :

```
CRTJRN  JRN(JRNLIB/DJRN1)
        JRNRCV(JRNLIB/RCV0001)
        MNGRCV(*SYSTEM) DLTRCV(*YES)
        TEXT('DataPropagator Journal')
```

- Indiquez le nom du récepteur de journal que vous avez créé à l'étape 1.
- Utilisez le paramètre Manage receiver (MNGRCV) pour que le système modifie le récepteur de journal et en associe un nouveau lorsque le récepteur associé devient trop gros. Si vous sélectionnez cette option, vous n'avez pas besoin d'utiliser la commande **CRTJRN** pour détacher les récepteurs et créer et associer manuellement de nouveaux récepteurs.
- Utilisez l'attribut par défaut MINENTDTA(\*NONE). Les autres valeurs ne sont pas valides pour ce mot clé.
- Indiquez DLTRCV(\*NO) uniquement si vous avez des raisons impérieuses de le faire (par exemple, si vous devez enregistrer ces récepteurs de journal pour des raisons de récupération). Si vous indiquez DLTRCV(\*YES), ces récepteurs pourraient être supprimés avant que vous ayez la possibilité de les enregistrer.

Vous pouvez utiliser ces deux valeurs sur le paramètre RCVSIZOPT de la commande **CRTJRN** (\*RMVINTENT et \*MINFIXLEN) pour optimiser votre

disponibilité de stockage et vos performances système. Voir le Guide *System i Programming: Performance Tools Guide* pour plus d'informations.

3. Commencez la journalisation de la table source à l'aide de la commande Start Journal Physical File (**STRJRPF**), comme dans l'exemple suivant :

```
STRJRPF FILE(library/file)
         JRN(JRNLIB/DJRN1)
         OMTJRNE(*OPNCLO)
         IMAGES(*BOTH)
```

Indiquez le nom du journal que vous avez créé au cours de l'Étape 2. Le programme Capture exige une valeur **\*BOTH** pour le paramètre **IMAGES**.

4. Modifiez la configuration de la journalisation de la table source :
  - a. Utilisez **IMAGES(\*BOTH)** pour vous assurer que la table source est journalisée pour les images avant et après.
  - b. Veillez à ce que le journal ait les attributs suivants : **MNGRCV(\*SYSTEM)** et **DLTRCV(\*YES)**.
  - c. Veillez à ce que le journal ait l'attribut **MINENTDTA(\*NONE)**.
  - d. Pour les journaux sur des systèmes distants, indiquez les attributs **MNGRCV(\*SYSTEM)**, **DLTRCV(\*YES)**, et **MINENTDTA(\*NONE)** dans le journal source. Pour définir le journal distant, indiquez l'attribut **DLTRCV(\*YES)** dans la commande **ADDRMTJRN**.

## Gestion des journaux et récepteurs de journal (System i)

Le programme Capture utilise la commande Receive Journal Entry (**RCVJRNE**) pour recevoir des journaux.

Les rubriques suivantes décrivent comment gérer les journaux et récepteurs de journal.

### Spécification de la gestion de système des récepteurs de journal (System i) :

Vous devez laisser le système System i gérer la modification des récepteurs de journal. C'est ce qu'on appelle la *gestion du journal de modifications du système*.

### Restrictions

Lorsque vous utilisez la commande **RTVJRNE** pour extraire les entrées de journal, un maximum de 299 fichiers source physiques peut utiliser le même journal et schéma de Capture. Si vous devez enregistrer plus de 299 fichiers dans le même journal, divisez vos enregistrements source en plusieurs schémas de Capture.

### Procédure

Pour spécifier la gestion de système des récepteurs de journal, indiquez **MNGRCV(\*SYSTEM)** lorsque vous créez le journal, ou modifiez le journal à cette valeur. Si vous utilisez le support de gestion du journal de modification du système, vous devez créer un récepteur de journal qui indique le seuil à partir duquel vous voulez que le système change les récepteurs de journal. Le seuil doit être d'au moins 5 000 Ko, et doit être basé sur le nombre de transactions de votre système. Le système détache automatiquement le récepteur lorsqu'il atteint la taille seuil et crée et associe un nouveau récepteur de journal s'il le peut.

### Journalisation distante avec des heures système différentes (System i) :

Si, dans un environnement de réplication qui utilise la journalisation distante, l'heure (QTIME) du système source et celle du système cible ne correspondent pas, certaines précautions doivent être prises lors de l'établissement de liaison initial entre les programmes Capture et Apply.

Lors de la réplication à l'aide de journaux distants, les programmes Capture et Apply agissent comme si le système source était en local. Dans la mesure du possible, faites correspondre les valeurs QTIME du système source et du système cible.

En cas d'impossibilité, prenez les précautions suivantes :

- Si le système source est en avance par rapport au système cible, installez la PTF SE23500/SI21622.
- S'il est en retard par rapport au système cible, le programme Capture commence par recevoir les modifications selon l'heure de la cible. Attendez que l'heure du système source ait dépassé l'heure de la régénération intégrale. Ensuite, faites des modifications test, puis vérifiez si elles ont été répliquées avant d'autoriser les applications à mettre à jour la table source.

En général, lorsque les heures des systèmes cible et source diffèrent, évitez toute opération entraînant une nouvelle régénération intégrale (utilisation des commandes CLRPFM et CPYF avec l'option \*REPLACE, par exemple).

Une fois que le programme Capture a commencé à traiter les modifications concernant la table source, vous pouvez définir la colonne DISABLE\_REFRESH de la table IBMSNAP\_REGISTER sur 1 pour cette table. Si une régénération intégrale est nécessaire, le programme Apply échoue ; vous pouvez alors coordonner la régénération intégrale lorsque vous êtes prêt à établir une liaison contrôlée, en définissant DISABLE\_REFRESH sur 0.

#### **Eviter les régénérations intégrales indésirables et autres problèmes engendrés par une non-concordance de temps (System i) :**

Lorsque les heures système des serveurs source et cible ne correspondent pas, des régénérations intégrales indésirables peuvent se produire et des entrées de journal peuvent manquer. Une procédure vous permet d'éviter ce genre de problème.

Les deux problèmes ont la même cause première. Lorsque le programme Apply signale qu'une régénération intégrale a démarré, un déclencheur sur le système où le programme Capture est en cours d'exécution remplit la valeur FR\_START\_TIME de la table IBMSNAP\_REG\_EXT (dans le cas du journal distant, le déclencheur est situé au niveau du système cible). L'heure que le déclencheur utilise est l'horodatage actuel de la machine où le déclencheur est en cours d'exécution. Le déclencheur n'est pas en mesure de récupérer l'horodatage du système source.

Les problèmes potentiels engendrés par les non-concordances de temps se répartissent en deux catégories et sont décrits dans les rubriques suivantes :

- «Horloge du système source en avance sur celle du système cible»
- «Horloge du système source en retard sur celle du système cible», à la page 42

#### **Horloge du système source en avance sur celle du système cible**

Lorsque l'horloge du système source est en avance sur celle du système cible, une commande Supprimer un membre de fichier physique (CLRPFM) pour la table source entraîne plusieurs régénérations intégrales.

Lorsqu'une commande CLRPFM (ou une entrée de journal similaire) est reçue, une régénération intégrale est démarrée. Dans les fonctions de journalisation à distance, l'horodatage du système cible est indiqué dans la zone FR\_START\_TIME. Si l'horloge du système source est en avance sur celle du système cible, la valeur FR\_START\_TIME sera inférieure à l'heure de la commande CLRPFM sur le système source.

Le programme Capture utilise FR\_START\_TIME comme heure à laquelle les entrées de journal doivent être collectées. Parce que la cible est en retard sur la source, Capture verra CLRPFM à nouveau, entraînant ainsi une nouvelle régénération intégrale. Cette boucle se répétera jusqu'à l'heure de la régénération intégrale sur le système cible se situe au-delà de l'heure de l'entrée de journal de CLRPFM.

Pour éviter ce problème, procédez comme suit :

1. Arrêtez le programme Apply mais continuez d'exécuter Capture.
2. Emettez la commande CLRPFM au niveau de la source.
3. Lorsque l'horodatage du système cible est ultérieur à l'heure d'émission de la commande CLRPFM sur le système source, démarrez le programme Apply.

### **Horloge du système source en retard sur celle du système cible**

Lorsque l'horloge du système cible est en avance sur celle du système source, Capture utilise FR\_START\_TIME afin de rechercher des entrées de journal ultérieures à celles attendues. Prenez en considération le scénario dans lequel le système cible est en avance d'une heure sur le système source :

- La régénération intégrale se produit à 05:00 sur le système cible, ce qui correspond à 04:00 sur le système source.
- Etant donné que FR\_START\_TIME est défini sur 05:00, le programme Capture ne trouve pas d'entrées de journal de la source jusqu'à ce que l'horloge du système source n'indique 05:00.
- Capture ne trouve pas les entrées de journal entre 04:00 et 05:00 sur le système source.

Pour éviter ce problème, procédez comme suit :

1. Assurez-vous que les programmes Capture et Apply sont arrêtés.
2. Démarrez Capture. Cette étape permet à Capture de prendre en compte les nouvelles tables qui attendent une régénération intégrale.
3. Attendez que les travaux de journalisation démarrent.
4. Arrêtez Capture et attendez que le programme se termine complètement.
5. Démarrez Apply et attendez la fin complète de la régénération intégrale.
6. Effectuez l'une des étapes suivantes :
  - Assurez-vous qu'aucune donnée n'aille dans la table source entre l'heure de fin de la régénération intégrale et l'heure à laquelle l'horloge du système source rattrape son retard sur celle du système cible.
  - Mettez à jour la valeur FR\_START\_TIME pour qu'elle corresponde à l'heure de l'horloge du système source lorsque la régénération intégrale a démarré.
7. Démarrez Capture.

### **Modification des définitions des objets de gestion des travaux (System i) :**

System i

Vous pouvez modifier les définitions des trois types d'objets de gestion des travaux créés pendant l'installation de DB2 DataPropagator pour System i, ou fournir vos propres définitions.

### A propos de cette tâche

Le programme d'installation crée un journal SQL, un récepteur de journal SQL pour cette bibliothèque, et des objets de gestion des travaux.

Le tableau 2 dresse la liste des objets qui sont créés.

Tableau 2. Objets de gestion des travaux

Description	Type d'objet	Nom
Description du sous-système	*SBSD	QDP4/QZSNDPR
File d'attente des travaux	*JOBQ	QDP4/QZSNDPR
Description du travail	*JOB	QDP4/QZSNDPR

Si vous créez votre propre définition du sous-système, vous devez nommer le sous-système QZSNDPR et le créer dans une bibliothèque différente de QDP4. Voir *System i Work Management Guide* (SC41-5306) pour plus d'informations sur la modification de ces définitions.

### Spécification de la gestion des utilisateurs des récepteurs de journal (System i) :

Si vous spécifiez MNGRCV(\*USER) lorsque vous créez le journal (au sens où vous voulez gérer les modification de vos propres récepteurs de journal), un message est envoyé à la file d'attente des messages du journal lorsque le récepteur atteint un seuil de stockage, si celui-ci a été spécifié pour le récepteur.

### A propos de cette tâche

Utilisez la commande **CHGJRN** pour détacher l'ancien récepteur du journal et en associer un nouveau. Cette commande empêche les conditions d'erreur Entrée non consignée et limite la quantité d'espace de stockage utilisée par le journal. Pour éviter d'affecter les performances, faites cela à un moment auquel le système n'est pas utilisé au maximum.

Vous pouvez faire rebasculer la gestion du récepteur de journal sur le système en spécifiant CHGJRN MNGRCV(\*SYSTEM).

Vous devez régulièrement détacher le récepteur de journal en cours et en associer un nouveau, pour deux raisons :

- L'analyse des entrées de journal est facilitée si chaque récepteur de journal contient des entrées pour une période spécifique et gérable.
- Les grands récepteurs de journal peuvent affecter les performances du système et occuper un espace précieux sur le stockage auxiliaire.

La file d'attente de messages par défaut pour un journal est QSYSOPR. Si vous avez un grand volume de messages dans la file d'attente de messages QSYSOPR, vous pouvez vouloir associer au journal une file d'attente de messages différente, telle que DPRUSRMSG. Vous pouvez utiliser un programme de gestion des messages pour contrôler la file d'attente des messages DPRUSRMSG. Pour une explication des messages qui peuvent être envoyés à la file d'attente des messages du journal, voir *System i Backup and Recovery*.

### Suppression d'une routine d'exit d'un récepteur de journal (System i) :

La routine d'exit *supprimer un récepteur de journal* (**DLTJRNRCV**) permet de s'assurer que les récepteurs de journal ne sont pas supprimés si le programme Capture n'a pas traité toutes les entrées qu'ils contiennent.

Lorsque vous installez DB2 DataPropagator pour System i, cette routine d'exit est automatiquement enregistrée. Elle est appelée à chaque fois qu'un récepteur de journal est supprimé, qu'il soit ou non utilisé pour la journalisation des tables source. Cette routine d'exit détermine si un récepteur de journal peut ou non être supprimé.

Pour tirer parti de la routine d'exit supprimer un récepteur de journal et laisser la gestion de journal au système, spécifiez **DLTRCV(\*YES)** et **MNGRCV(\*SYSTEM)** dans la commande **CHGJRN** ou **CRTJRN**.

**Avertissement :** Si vous supprimez l'enregistrement de la routine d'exit supprimer un récepteur de journal, vous devez modifier tous les journaux utilisés pour les tables source afin qu'ils aient l'attribut **DLTRCV(\*NO)**.

Si le journal associé au récepteur n'est pas associé à l'une des tables source, cette routine d'exit *approuve* la suppression du récepteur.

Si le récepteur de journal est utilisé par une ou plusieurs tables source, cette routine d'exit s'assure que le récepteur supprimé ne contient pas d'entrées n'ayant pas été traitées par le programme Capture. La routine d'exit *désapprouve* la suppression du récepteur si le programme Capture doit encore traiter des entrées sur ce récepteur.

Si vous devez supprimer un récepteur de journal et que la routine d'exit supprimer un récepteur de journal n'approuve pas la suppression, indiquez **DLTJRNRCV DLTLOPT(\*IGNEXITPGM)** pour remplacer la routine d'exit.



---

## Chapitre 4. Enregistrement de tables et de vues en tant que sources de réplication SQL

Grâce à la réplication SQL, vous pouvez identifier les tables et les vues à utiliser en tant que sources de réplication, en les enregistrant.

Lorsque vous enregistrez une table ou une vue spécifique à des fins de réplication, vous créez une source de données disponibles que vous pourrez utiliser ultérieurement avec différentes cibles et différents objectifs. Les tâches d'administration décrites dans cette section vous permettent de configurer les informations de contrôle qui définissent la façon dont les données sont capturées à partir de chaque source selon vos objectifs concernant la réplication.

Lorsque vous enregistrez une source, vous identifiez la table ou la vue à utiliser en tant que source de réplication, vous sélectionnez les colonnes que vous souhaitez utiliser pour la réplication, et les propriétés de capture des données et des modifications réalisées par le programme de réplication à partir de la source.

Pour la réplication SQL, vous pouvez enregistrer les objets suivants en tant que sources :

- Une table DB2
- Une table relationnelle non DB2 avec pseudonyme
- Un sous-ensemble des données d'une table(DB2 ou table relationnelle non DB2)
- Une vue de table(DB2)
- Une vue représentant une jointure interne de deux tables ou davantage (DB2)

---

### Enregistrement de tables DB2 en tant que sources

Lorsque vous enregistrez une table DB2 en tant que source de réplication, vous devez spécifier le serveur source, le nom de la table source et le schéma Capture. Une table de modification des données (CD) est créée.

#### Avant de commencer

- Pour toutes les sources DB2 à l'exception de System i, la table source DDL nécessite l'utilisation de l'option DATA CAPTURE CHANGES. Ne supprimez pas cette option de votre source.
- Les tables de contrôle Capture doivent se trouver sur le serveur de contrôle Capture qui traitera la table à enregistrer en tant que source.

#### Restrictions

##### System i

- La longueur des instructions SQL est limitée à 32 000 caractères ; par conséquent, vous ne pouvez enregistrer qu'environ 2000 colonnes par vue (le nombre exact de colonnes dépend de la longueur des noms de colonne).
- Pour le schéma Capture, n'enregistrez pas plus de 300 tables source utilisant le même journal.
- Les tables source, les tables CD et les journaux de vos tables source doivent tous figurer dans le même fichier ACF (Auxiliary Storage Pool)

que les tables de contrôle Capture qui contiennent les informations d'enregistrement relatives à ces tables source.

#### 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.

### A propos de cette tâche

La réplication SQL prend en charge les types suivants de tables DB2 en tant que sources :

#### z/OS

- Tables DB2 gérées par votre application
- Tables de catalogue
- Tables CCD externes

#### System i

- Tables DB2 gérées par votre application (consignées localement ou à distance)
- Tables CCD externes

#### Linux UNIX Windows

- Tables DB2 gérées par votre application
- Tables de catalogue (pour la réplication avec régénération intégrale uniquement)
- Tables de requêtes matérialisées
- Tables CCD externes
- Les tables partitionnées avec la clause `DISTRIBUTE BY` de l'instruction `CREATE TABLE`
- Les tables partitionnées par intervalle (avec la clause `PARTITION BY` de l'instruction `CREATE TABLE`)
- Les tables compressées

Vous pouvez enregistrer la même table plusieurs fois à l'aide de différents schémas Capture.

### Procédure

Pour enregistrer une table DB2, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Programme de ligne de commande ASNCLP</b>	Utilisez la commande <b>CREATE REGISTRATION</b> pour enregistrer une table source, une vue ou un pseudonyme. Par exemple, les commandes suivantes définissent l'environnement et enregistrent la table STAFF dans la base de données DB2 SAMPLE.  SET SERVER CAPTURE TO DB SAMPLE; SET OUTPUT CAPTURE SCRIPT "register.sql"; SET LOG "register.err"; SET RUN SCRIPT LATER; CREATE REGISTRATION (DB2ADMIN.STAFF) DIFFERENTIALREFRESH STAGE CDSTAFF;

Méthode	Description
Centre de répliation	Utilisez la fenêtre Enregistrer les tables. Dans l'arborescence d'objets, développez le schéma Capture de votre choix, cliquez avec le bouton droit de la souris sur le dossier <b>Tables enregistrées</b> , puis cliquez sur <b>Enregistrer les tables</b> . <b>Conseil :</b> Pour gagner du temps, vous pouvez créer un profil d'objet source que le serveur de contrôle Capture utilisera. Lorsque vous enregistrez une table, le Centre de répliation utilise dans ce cas les valeurs par défaut définies dans le profil d'objet source, et non les valeurs par défaut du Centre de répliation. Cela vous permet d'accélérer la procédure d'enregistrement, car vous remplacez les valeurs par défaut une seule fois au lieu de sélectionner les tables une par une et de modifier manuellement les valeurs par défaut.
<div style="background-color: #800000; color: white; padding: 2px; text-align: center; font-weight: bold;">System i</div> Commande système ADDDPRREG	Utilisez la commande ADDDPRREG pour enregistrer une vue sous System i.  Par exemple, pour enregistrer la table source EMPLOYEE de la bibliothèque HR dans le schéma Capture et créer la table CD CDEMPLOYEE dans la bibliothèque HRCDLIB :  ADDDPRREG SRCTBL(HR/EMPLOYEE) CAPCTLLIB(BSN) CDLIB(HRCDLIB) CDNAME(CDEMPLOYEE)

Lorsque vous enregistrez une table en tant que source, le programme Capture associé à la table enregistrée lit le journal de la source et stocke en mémoire les modifications apportées aux colonnes enregistrées jusqu'à ce que la transaction soit validée ou annulée. Pour les annulations, les modifications sont supprimées de la mémoire. Pour les validations, les modifications sont insérées dans la table CD dès que le programme Capture lit l'enregistrement de journal de validation. Ces modifications sont conservées en mémoire jusqu'à ce que le programme Capture les valide après chaque cycle Capture. Le programme Capture ne commence pas à capturer les données pour une table source DB2 tant qu'un signal CAPSTART n'a pas été émis (par vous ou par le programme Apply).

**Pour les tables de source non relationnelles :** Vous pouvez enregistrer les tables DB2 qui contiennent des données issues de systèmes de gestion de bases de données non relationnelles, tels que IMS. Pour cela, vous devez utiliser une application telle que IMS,DataPropagator ou Data Refresher, qui fournit une table CCD contenant les données issues de la base de données non relationnelle. L'application capture les modifications dans les segments non relationnels de la base de données IMS et remplit une table CCD. la table CCD doit être complète, mais elle peut être condensée ou non condensée. Comme pour les autres sources CCD, un programme Capture est associé, ainsi qu'une table source CCD, car la table stocke déjà les données modifiées issues de la table source non relationnelle. Les produits IMS,DataPropagator et Data Refresher permettent de gérer les valeurs contenues dans la table IBMSNAP\_REGISTER ; le programme Apply peut donc effectuer une lecture correcte dans cette table source.

---

## Enregistrement de tables relationnelles non DB2 en tant que sources

Lorsque vous enregistrez une table relationnelle non DB2, vous spécifiez le pseudonyme de la table source à enregistrer. Une table de modification cohérente des données (CCD) est alors créée.

### Avant de commencer

Les tables de contrôle Capture doivent se trouver sur le serveur de contrôle Capture qui effectuera cette opération.

### Restrictions

- Si vous utilisez une base de données fédérée pour l'accès aux serveurs de sources de données relationnelles non DB2, vous devez utiliser un schéma Capture différent pour chacun d'entre eux avec cette base de données fédérées. Deux schémas ne peuvent pas être identiques. Vous pouvez enregistrer une table relationnelle non DB2 via un seul schéma Capture.
- Vous ne pouvez pas enregistrer les colonnes de valeurs LOB des tables relationnelles non DB2. Si vous enregistrez une table qui contient ce type de données, vous devez enregistrer un sous-ensemble de colonnes.

### A propos de cette tâche

Par défaut, le nom du propriétaire d'une table CCD est dérivé du nom de schéma de la table source. Si vous modifiez le nom du propriétaire de table CCD afin qu'il ne corresponde pas au nom du schéma, assurez-vous que le propriétaire de la table source possède des droits en écriture sur la table CCD. Si le propriétaire de la table source ne peut pas mettre à jour la table CCD, les déclencheurs de cette table ne pourront pas enregistrer les modifications dans la table CCD.

### Procédure

Pour enregistrer une table relationnelle non DB2, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande <b>CREATE REGISTRATION</b> pour enregistrer une table source, une vue ou un pseudonyme. Par exemple, les commandes suivantes définissent l'environnement et créent un enregistrement dont les caractéristiques sont les suivantes :</p> <ul style="list-style-type: none"> <li>• Serveur non IBM contenant la base de données Oracle V9ORA</li> <li>• Serveur fédéré FEDORADB</li> <li>• Table CCD de base de données Oracle undjr09.ccdtest</li> <li>• Pseudonyme CCD du serveur fédéré repldba.ccdtestnk</li> <li>• Pseudonyme source enregistré repldba.tesnk</li> <li>• Toutes les colonnes présentes dans repldba.tesnk sont enregistrées avec des images après</li> </ul> <pre>SET SERVER CAPTURE TO DB FEDORADB NONIBM SERVER V9ORA ID repldba PASSWORD "passwd"; SET OUTPUT CAPTURE SCRIPT "ora_reg.sql"; SET CAPTURE SCHEMA SOURCE ASNORA; SET LOG "orareg.out"; CREATE REGISTRATION (repldba.testnk) DIFFERENTIALREFRESH STAGE repldba.ccdtestnk CONDENSED OFF NONIBM undjr09.ccdtest COLS ALL IMAGE AFTER;</pre> <p>L'option CONDENSED OFF est requise pour les sources fédérées.</p>

Méthode	Description
Centre de réplication	Utilisez la fenêtre Enregistrer les pseudonymes. Dans l'arborescence d'objets, développez la base de données relationnelles non DB2 contenant les pseudonymes à enregistrer. Cliquez avec le bouton droit de la souris sur le dossier <b>Pseudonymes</b> et sélectionnez <b>Enregistrer les pseudonymes</b> . <b>Conseil :</b> Pour gagner du temps, vous pouvez créer un profil d'objet source que le serveur de contrôle Capture utilisera. Lorsque vous enregistrez une table, le Centre de réplication utilise dans ce cas les valeurs par défaut définies dans le profil d'objet source et les pseudonymes associés aux tables CCD, et non les valeurs par défaut du Centre de réplication. Cela vous permet d'accélérer la procédure d'enregistrement, car vous remplacez les valeurs par défaut une seule fois au lieu de sélectionner les tables une par une et de modifier manuellement les valeurs par défaut.

Lorsqu'une modification de table relationnelle non DB2 enregistrée est effectuée, les déclencheurs Capture simulent le programme Capture et insèrent la modification dans la table CCD. Lorsque vous enregistrez la source, les déclencheurs Capture commencent à capturer les modifications d'une table source relationnelle non DB2.

---

## Options d'enregistrement applicables aux tables source

La réplication SQL offre un grand nombre d'options que vous pouvez utiliser pour l'enregistrement d'une table en tant que source de réplication. Ces options font partie de la tâche d'enregistrement d'une table.

Une fois que vous avez choisi la table à enregistrer, vous pouvez identifier les colonnes à utiliser pour la réplication, et vous pouvez définir les propriétés qui déterminent le traitement et le stockage des données enregistrées. Vous pouvez également spécifier d'autres options de réplication, telles que le mode de stockage des données source dans la table de modification des données (CD) par le programme Capture, ou le mode de stockage des données dans la table CCD réalisé par les déclencheurs Capture.

### Enregistrement d'un sous-ensemble de colonnes (vertical)

Vous pouvez enregistrer un sous-ensemble des colonnes de la table source à des fins de réplication, par exemple si vous ne souhaitez pas que toutes les colonnes disponibles pour les cibles s'abonnent ou si les tables cible ne prennent pas en charge tous les types définis pour la table source.

Par défaut, toutes les colonnes sont enregistrées. Pour enregistrer un sous-ensemble de colonnes, sélectionnez uniquement les colonnes que vous souhaitez rendre admissibles pour la réplication dans une table cible.

Les tables CD et CCD doivent contenir une quantité suffisante de données clés pour certains types de tables cible (point de cohérence, par exemple) ; par conséquent, assurez-vous que votre sous-ensemble contient les colonnes qui seront les colonnes clés (clé primaire ou index unique) pour la cible.

**Conseil :** N'enregistrez un sous-ensemble des colonnes source que si vous êtes certain que vous ne souhaitez jamais répliquer les colonnes non enregistrées. Si vous souhaitez répliquer ultérieurement les colonnes que vous n'avez pas enregistrées, vous devez modifier vos enregistrements afin d'ajouter ces colonnes

non enregistrées. (Pour les sources relationnelles non DB2, vous devez redéfinir vos enregistrements afin d'ajouter de nouvelles colonnes à un enregistrement.) Si vous envisagez d'associer une table CCD interne à cette source, cela peut s'avérer encore plus difficile d'ajouter des colonnes, car l'enregistrement de nouvelles colonnes ajoute celles-ci à la table CD, mais non à la table CCD interne. Pour éviter ces incidents, vous pouvez enregistrer toutes les colonnes et utiliser le programme Apply pour établir des sous-ensembles de colonnes répliquées sur des cibles.

## Réplication en mode capture des modifications et régénération intégrale

Par défaut, seules les modifications apportées à la table source depuis le dernier cycle de réplication sont répliquées (réplication en mode capture des modifications). Vous pouvez également répliquer toutes les données de la table source au cours de chaque cycle (réplication avec régénération intégrale uniquement).

### Réplication en mode capture des modifications

Au cours de la réplication en mode capture des modifications, seules les données modifiées sont répliquées dans la table cible. En fonction du type de table cible sélectionné pour cette source, vous devez effectuer un chargement initial de la table. Dans la plupart des cas, le programme Apply effectue une régénération intégrale initiale, puis poursuit en exécutant une réplication en mode capture des modifications.

Si vous avez choisi de ne pas autoriser la régénération intégrale des tables cible, vous devez recharger manuellement la table si les tables source et cible doivent être resynchronisées. Une fois la cible chargée avec les données source initiales, le programme Capture effectue la capture des modifications apportées sur la source, puis les stocke dans la table de modification des données (CD). Lors de la réplication en mode capture des modifications pour sources relationnelles non DB2, les déclencheurs Capture effectuent la capture des modifications sur la source et les stocke dans la table de modification cohérente des données (CCD). Le programme Apply lit les modifications de la table CD ou CCD, puis les applique aux cibles abonnées à la source enregistrée.

Lorsque vous définissez une table source DB2 pour la réplication en mode capture des modifications, vous ne souhaitez pas toujours stocker toutes les modifications réalisées sur la source dans la table CD. Vous pouvez enregistrer un sous-ensemble de lignes (horizontales) qui filtre les modifications ; cela permet de capturer un nombre de modifications plus réduit qu'au niveau de la source. Vous pouvez sélectionner une règle de capture de lignes parmi les deux règles suivantes, afin de déterminer quelles lignes modifiées de la table source le programme Capture doit enregistrer dans la table CD :

- Les modifications apportées à toutes les lignes sont capturées.
- Les modifications ne sont capturées que si elles concernent une colonne enregistrée. (DB2 uniquement)

Par défaut, les modifications sont capturées chaque fois qu'une ligne est mise à jour pour une colonne (enregistrée ou non), dans la table source. Si vous enregistrez uniquement un sous-ensemble de colonnes, le programme Capture enregistre les valeurs de ligne des colonnes enregistrées de la table CD, chaque fois qu'une modification est apportée à la table source (même si les colonnes modifiées diffèrent des colonnes enregistrées). Utilisez cette option par défaut si vous désirez conserver un historique de toutes les modifications apportées à la table source. Il

s'agit de l'unique option disponible pour les sources relationnelles non DB2 ; les déclencheurs Capture effectuent la capture de toutes les lignes modifiées sur la source, même si la modification a lieu dans une colonne non enregistrée.

**Exemple :** supposons que votre table contienne 100 colonnes et que vous enregistreriez 50 de ces colonnes en vue de la réplication. Par défaut, chaque fois qu'une modification est apportée à l'une des 100 colonnes de votre table, le programme Capture enregistre une ligne dans la table CD (les déclencheurs Capture peuvent également enregistrer une ligne dans la table CCD).

Si vous disposez d'une source DB2, le programme Capture peut effectuer la capture des modifications pour colonnes enregistrées uniquement. Dans ce cas, le programme Capture enregistre une ligne dans la table CD uniquement lorsque des modifications sont apportées à des colonnes enregistrées.

**Conseil :** Choisissez de capturer les modifications de toutes les lignes si vous avez besoin d'informations à des fins d'audit, ou si les modifications apportées à la table ont presque toujours lieu dans des colonnes enregistrées. Choisissez de capturer les modifications pour les colonnes non enregistrées uniquement si des modifications fréquentes affectent uniquement des colonnes non enregistrées. Utilisez cette option si vous ne désirez pas conserver un historique de toutes les modifications apportées à la table source.

## Réplication avec régénération intégrale uniquement

Lorsque des cibles sont abonnées à une source enregistrée pour la réplication avec régénération intégrale, le programme Apply supprime toutes les données de la table cible, copie les données situées dans les colonnes enregistrées de la source, et remplit les cibles à l'aide des données source au cours de chaque cycle de réplication. Le programme Capture n'est pas concerné, et il n'y a pas de table CD ; le programme Apply lit directement les données à partir de la source.

### Tables peu volumineuses

Si votre table est peu volumineuse et si sa copie ne consomme que peu de temps ou de ressources, il est inutile de choisir la réplication avec régénération intégrale.

### Tables volumineuses

Si vos tables sont volumineuses et que vous souhaitez utiliser la réplication avec régénération intégrale, il est conseillé d'utiliser la routine d'exit ASNLOAD pour accélérer le chargement de vos tables.

**Restriction :** Si vous souhaitez utiliser une table cible condensée abonnée à cette source et que vous ne parvenez pas à insérer un index unique dans cette table cible, vous devez enregistrer la source pour une réplication avec régénération intégrale uniquement.

## Colonnes d'image après et d'image avant

Lorsque vous enregistrez une source pour la réplication en mode capture des modifications, par défaut seule la valeur modifiée (image après) d'une colonne est capturée. Vous pouvez également choisir de capturer la valeur précédente (image avant).

z/OS

Linux UNIX Windows

Vous pouvez choisir de capturer les valeurs d'image avant pour les différentes colonnes d'une table.

Vous pouvez choisir de capturer les valeurs d'image avant pour les différentes colonnes d'une table, ou encore pour aucune d'entre elles. Vous ne pouvez pas sélectionner cette option pour chaque colonne individuellement.

#### **Sybase ou Microsoft SQL Server**

Une table ne peut contenir qu'une colonne de type `TIMESTAMP`. Lorsque la source de données est Sybase ou Microsoft SQL Server et que la table source contient une colonne de type `TIMESTAMP`, vous ne devez sélectionner que les images après de cette colonne lorsque vous la définissez comme faisant partie intégrante de la source de réplication.

**Restriction :** Vous ne pouvez pas inclure de valeurs d'image avant dans la table de modification des données (CD) pour les colonnes portant des types de données LOB.

Les sections ci-dessous indiquent à quel moment vous devez choisir chaque option.

### **Capture de nouvelles valeurs image après uniquement**

Pour chaque colonne que vous enregistrez à des fins de réplication en mode capture des modifications, vous pouvez spécifier, pour chaque modification, qu'un enregistrement de la valeur d'image après uniquement soit effectué par le programme ou les déclencheurs Capture. Lorsque vous choisissez de capturer les valeurs d'image après uniquement, la table CD (ou CCD) inclut une colonne par valeur modifiée (qui stocke la valeur de la colonne source une fois la modification apportée).

Vous n'avez pas besoin des images avant si vous envisagez d'utiliser uniquement pour cette source les types de table cible d'agrégats de base et d'agrégats de modification. Les colonnes d'images avant sont inutiles si vous souhaitez utiliser votre table cible pour les valeurs calculées, car il n'existe pas d'images avant pour les colonnes calculées. Tous les autres types de tables cible peuvent utiliser les colonnes d'images avant.

### **Capture de valeurs d'image avant et d'image après**

Pour chaque colonne que vous enregistrez à des fins de réplication en mode capture des modifications, vous pouvez spécifier, pour chaque modification, qu'un enregistrement de la valeur d'image avant et de la valeur d'image après soit effectué par le programme ou les déclencheurs Capture. Lorsque vous choisissez de capturer les valeurs d'image avant et après, la table CD (ou CCD) inclut deux colonnes par valeur modifiée : une colonne pour la valeur de la colonne source avant la modification et une colonne pour la valeur après modification.

Lorsque vous choisissez de stocker à la fois les images avant et les images après dans la table CD (ou CCD), les colonnes d'image avant et d'image après portent des valeurs différentes pour les différentes opérations effectuées dans les tables source :

#### **Insertion**

La colonne d'image avant contient une valeur `NULL`. La colonne d'image après contient la valeur insérée.



### **Mise à jour**

La colonne d'image avant contient la valeur de colonne avant l'application de la modification. La colonne d'image après contient la valeur de colonne après la modification.

Lorsque vous choisissez de capturer les mises à jour sous forme de couples de suppressions et d'insertions, la ligne de suppression contient l'image avant de la mise à jour dans les colonnes image avant et image après de la ligne, et la ligne d'insertion contient des valeurs NULL dans la colonne image avant et dans la colonne image après.

### **Suppression**

Les colonnes d'image avant et d'image après contiennent la valeur de colonne avant l'application de la modification.

Pour les colonnes contenant des images avant définies, la réplication limite le nombre de caractères des noms de colonnes à 127, car le nom de la colonne ne peut pas dépasser 128 caractères. Si le nom de colonne est plus long, la réplication tronque les caractères supplémentaires à droite par défaut, sauf si vous avez défini un profil spécifiant une troncature à gauche. La réplication ajoute un identificateur de colonne d'image avant (X, par défaut) aux colonnes cible et chaque nom de colonne doit être un nom unique ; par conséquent, vous ne pouvez pas utiliser de noms de colonnes dépassant 127 caractères. Pour les tables que vous ne souhaitez pas répliquer, vous pouvez utiliser des noms de colonnes plus longs, mais il est conseillé de ne pas dépasser 127 caractères pour le cas où vous souhaiteriez répliquer ces colonnes ultérieurement.

La liste suivante décrit les situations dans lesquelles vous pouvez envisager de capturer les valeurs d'image avant :

#### **Pour conserver un historique de vos données source**

Si vous souhaitez conserver un historique de vos données à des fins d'audit, vous pouvez sélectionner les images avant et les images après, afin de conserver un enregistrement des modifications apportées sur une période donnée. Disposer de copies d'images avant et d'images après est utile dans certains secteurs d'activité qui effectuent des audits ou qui utilisent des fonctions d'annulation d'applications.

#### **Pour les configurations de réplication bidirectionnelle avec détection de conflit**

Dans les configurations de réplication bidirectionnelle dans lesquelles des conflits peuvent intervenir entre tables réplique (lorsque la détection de conflit porte une autre valeur que la valeur Aucune), vous devez enregistrer à la fois les colonnes d'image avant et d'image après de la table CD des répliques, afin que les modifications puissent être annulées en cas de conflit.

#### **Lorsque les colonnes clés de la cible sont susceptibles d'être mises à jour**

Quand vous enregistrez une source, réfléchissez aux tables cible potentielles que vous pouvez définir à l'aide de cette table comme source. En général, les tables cible sont condensées et nécessitent une colonne ou un ensemble de colonnes qui rendent uniques les lignes de cette table cible. Ces colonnes uniques forment ce que l'on appelle la clé cible. Si l'une de ces colonnes de clés cible est susceptible d'être modifiée au niveau de la source, la réplication SQL nécessite l'application d'un traitement spécifique pour garantir que les lignes mises à jour dans la table cible sont les lignes appropriées. Pour garantir que la réplication SQL met à jour les lignes appropriées avec la nouvelle valeur de clé dans la table cible, vous pouvez choisir de capturer les images-avant et les images-après pour les colonnes

qui constitueront la clé cible. Le programme Apply a besoin des valeurs d'image avant de ces colonnes enregistrées lorsqu'il applique les modifications apportées aux colonnes autres que les colonnes clés aux colonnes clés cible de la table cible. Lorsqu'il applique les modifications, le programme Apply recherche la ligne appropriée dans la table cible en tentant de localiser les valeurs clés cible correspondant à la valeur d'image avant de la table CD (ou CCD) source, puis il met à jour cette ligne cible à l'aide de la valeur d'image après de la table CD (ou CCD) source.

Même si vous enregistrez ces valeurs d'image avant au moment de l'enregistrement de la table ou vue source, la réplication ne sait pas que votre application effectuera des mises à jour de la clé cible. Lorsque vous définirez ultérieurement les cibles abonnées à cette source (via la création d'ensembles d'abonnements), vous pourrez demander au programme Apply d'effectuer des mises à jour spécifiques lorsque vous appliquerez des modifications de colonnes non clés de la source à des colonnes clés de la cible.

## Préfixe image avant

Si vous capturez des colonnes image après et image avant, la colonne image après prend le nom de la colonne de la table source, et l'image avant prend le nom de la colonne de la table source auquel vient s'ajouter un préfixe d'un caractère.

Le préfixe par défaut image-avant affecté par le programme de ligne de commande ASNCLP et par le Centre de réplication est X. La valeur par défaut des commandes système System i est @.

Vous pouvez modifier le préfixe par défaut. L'association du préfixe d'image avant et du nom de colonne de table CD ou CCD ne peut pas être identique à un nom de colonne existant ou potentiel dans la table CD ou CCD.

**Exemple :** si vous utilisez X en tant que préfixe d'image avant et que vous enregistrez une colonne source appelée COL, vous ne pouvez pas enregistrer la colonne XCOL car on ne sait pas exactement si XCOL est le nom de colonne réel d'une autre colonne source, ou encore le nom d'une colonne d'image avant (COL) portant le préfixe X.

**Restriction :** Vous ne pouvez pas utiliser un caractère blanc en tant que préfixe d'image avant.

Si vous ne répliquez pas de colonnes d'images avant pour une table, vous pouvez choisir de ne pas ajouter de préfixe et affecter la valeur Null à cette propriété.

## Arrêt du programme Capture après une erreur

Lorsque le programme Capture détecte certains incidents pendant le traitement des enregistrements, il s'arrête (comportement par défaut). Vous pouvez choisir de laisser le programme fonctionner.

La liste ci-dessous contient des informations qui peuvent vous aider à choisir la meilleure option pour votre environnement.


### Arrêter Capture après une erreur

Quand cette option est activée, le programme Capture enregistre un message d'erreur dans la table IBMSNAP\_CAPTRACE et s'arrête.

Le programme Capture s'arrête lorsque les erreurs bloquantes suivantes se produisent :

- La table de modification des données (CD) est saturée.
- L'erreur SQLCODE-911 se produit 10 fois de suite.
- Des erreurs SQL imprévues se produisent.

Le programme Capture ne s'arrête pas lorsque certaines erreurs non bloquantes se produisent, telles que :

- SQLCODES indique une longueur de données non valide.
-  Le dictionnaire de compression n'existe pas.

Lorsque des erreurs non bloquantes se produisent, le programme Capture invalide les enregistrements et continue de fonctionner.

### **Ne pas arrêter Capture après une erreur**

L'exécution du programme Capture se poursuit lorsque certaines erreurs se produisent. S'il rencontre des erreurs pendant la première tentative de traitement de la source, il n'active pas l'enregistrement. Si la source enregistrée était déjà active, il arrête le traitement de l'enregistrement. Dans les deux cas, l'enregistrement est arrêté. Un enregistrement qui a été arrêté porte la valeur "S" (arrêté) dans la colonne STATE de la table de contrôle IBMSNAP\_REGISTER.

Cette option n'arrête pas le programme Capture lorsque les erreurs non bloquantes suivantes se produisent :

- L'enregistrement n'est pas défini correctement.
- Le programme Capture n'a pas trouvé la table de modification des données (CD) lorsqu'il a tenté d'insérer des lignes pour les données modifiées.
- L'option DATA CAPTURE CHANGES de la table source (non-System i) a été détectée comme étant désactivée au démarrage du programme Capture, ou lors de sa réinitialisation.

Si l'état d'enregistrement d'un membre d'ensemble d'abonnements est l'état d'arrêt suite à une erreur, le programme Apply ne pourra pas traiter l'ensemble.

## **Options de stockage des mises à jour par le programme Capture**

Par défaut, les mises à jour apportées à la table source sont stockées sur une seule ligne de la table de modification des données. Dans certains cas, vous devez indiquer au programme Capture ou aux déclencheurs qu'il est nécessaire de déclencher la capture des mises à jour sous la forme de couples DELETE et INSERT enregistrés sur deux lignes.

Vous devez capturer les mises à jour sous forme d'instructions DELETE et INSERT lorsque vos applications source mettent à jour une ou plusieurs colonnes référencées par un prédicat au niveau d'un membre d'ensemble d'abonnements.

**Exemple :** Supposons que vous envisagiez de définir une cible abonnée uniquement à des données source possédant un prédicat qui prend comme base une valeur de colonne spécifique (WHERE DEPT = 'J35', par exemple). Lorsque vous modifiez cette colonne (en spécifiant DEPT='FFK', par exemple), la modification capturée n'est pas sélectionnée à des fins de réplication vers la cible, car elle ne correspond pas aux critères du prédicat. Cela signifie que le nouveau

service FFK ne sera pas répliqué, car le membre d'ensemble d'abonnements prend comme base le service J35. Lorsque vous convertissez les mises à jour en couple DELETE-INSERT, la ligne de table cible est supprimée.

Chaque mise à jour capturée est convertie en deux lignes dans la table de modification des données (ou table CCD), pour toutes les colonnes. Vous devez peut-être ajuster l'allocation d'espace au niveau de la table de modification des données (ou table CCD) afin de tenir compte de cette augmentation des données capturées.

## **Comment éviter de recapturer les modifications (réplication bidirectionnelle uniquement)**

Pour la réplication bidirectionnelle, vous pouvez utiliser l'option de recapture afin de contrôler si des modifications répliquées sur un site sont recapturées sur le second site à des fins de réplication sur d'autres sites.

**Restriction :** Les tables issues de bases de données relationnelles non DB2 ne peuvent pas être utilisées pour les réplifications bidirectionnelles. Cette option ne s'applique qu'aux sources de données DB2.

Dans la réplication bidirectionnelle, les changements peuvent provenir de la table maître ou des tables réplique associées. Lorsque vous enregistrez une table à utiliser dans la réplication bidirectionnelle, la réplication SQL suppose que cette table constitue la table maître au sein de votre configuration.

Pendant l'enregistrement, vous définissez l'option de recapture de la table maître. Ensuite, lorsque vous effectuez le mappage entre la table source et les tables réplique cible, vous pouvez déterminer si les modifications apportées à la réplique doivent être recapturées et transmises aux autres tables.

Lorsque vous enregistrez la table source qui sera la table maître de votre configuration de réplication bidirectionnelle, vous avez le choix entre les deux options suivantes :

### **Recapturer les modifications au niveau de la table maître**

Cette option permet de recapturer les mises à jour apportées à la table maître à partir d'une table réplique et de les retransmettre à d'autres répliques.

### **Ne pas recapturer les modifications au niveau de la table maître**

Cette option permet de ne pas recapturer les mises à jour apportées à la table maître à partir d'une table réplique et de ne pas les retransmettre à d'autres répliques.

Lorsque vous enregistrez la table réplique dans votre configuration de réplication bidirectionnelle, vous avez le choix entre les deux options suivantes :

### **Recapturer les modifications au niveau de la réplique**

Cette option permet de recapturer les mises à jour apportées à la table réplique à partir de la table maître et de les retransmettre à d'autres répliques abonnées à cette table réplique.

### **Ne pas recapturer les modifications au niveau de la réplique**

Cette option permet de ne pas recapturer les mises à jour apportées à la table réplique à partir de la table maître et de ne pas les retransmettre à d'autres répliques abonnées à cette table réplique.

Lorsque vous empêchez la recapture des modifications, vous pouvez augmenter les performances et diminuer les coûts de stockage : en effet, cela évite au programme Capture d'effectuer une recapture des mêmes modifications pour chaque réplique.

Les rubriques suivantes décrivent les méthodes à utiliser pour déterminer si une recapture des modifications doit avoir lieu en fonction de votre configuration de réplication bidirectionnelle.

### **Tables maître contenant une seule réplique**

Si vous envisagez d'inclure une seule réplique dans votre configuration de réplication bidirectionnelle, créez votre enregistrement de telle sorte que les modifications ne soient pas recapturées au niveau de la table maître ou de la table réplique.

Cette configuration est optimale si la table maître n'est pas la source d'autres tables réplique et que la réplique ne représente pas la source d'autres répliques (au sein d'une configuration multiniveau). Si seules ces deux tables sont concernées, les modifications effectuées dans la table réplique ne doivent pas obligatoirement être recapturées dans la table maître ; inversement, les modifications apportées à la table maître ne doivent pas obligatoirement être recapturées dans la table réplique.

### **Répliques multiples représentant des partitions mutuellement exclusives de la table maître**

Pour les répliques multiples qui représentent des partitions mutuellement exclusives de la table maître, créez votre enregistrement de telle sorte que les modifications ne soient pas recapturées au niveau de la table maître ou des tables réplique.

Si vous envisagez d'utiliser plusieurs répliques qui sont des partitions de la table maître, vous pouvez empêcher la recapture des modifications au niveau de la table maître et de chaque réplique. Cette configuration est optimale si aucune des répliques ne représente la source d'autres tables réplique. Lorsque les répliques sont des partitions de la table maître, deux répliques ne peuvent pas être abonnées aux mêmes données de la table maître. Par conséquent, les modifications effectuées dans la table réplique ne doivent pas obligatoirement être recapturées dans la table maître et transférées vers les autres répliques, car seule la réplique dans laquelle la modification a été apportée est abonnée à ces données source.

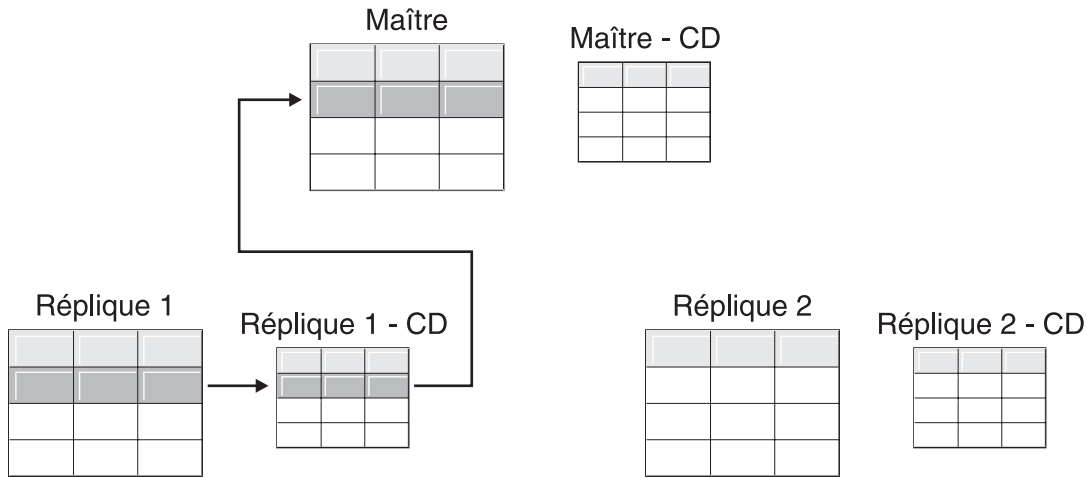


Figure 1. Option de recapture pour les répliques qui sont des partitions mutuellement exclusives de la table maître. Lorsque plusieurs répliques ne sont pas abonnées aux mêmes données dans la table maître, il est inutile d'utiliser l'option de recapture dans les tables.

### Tables maître répliquant les modifications dans plusieurs répliques

Pour les tables maître qui répliquent les modifications dans plusieurs répliques, créez votre enregistrement de telle sorte que les modifications soient recapturées dans la table maître mais non recapturées dans les tables réplique.

Les modifications provenant d'une réplique sont ensuite recapturées dans la table maître et répliquées dans les autres répliques abonnées aux données maître mises à jour.

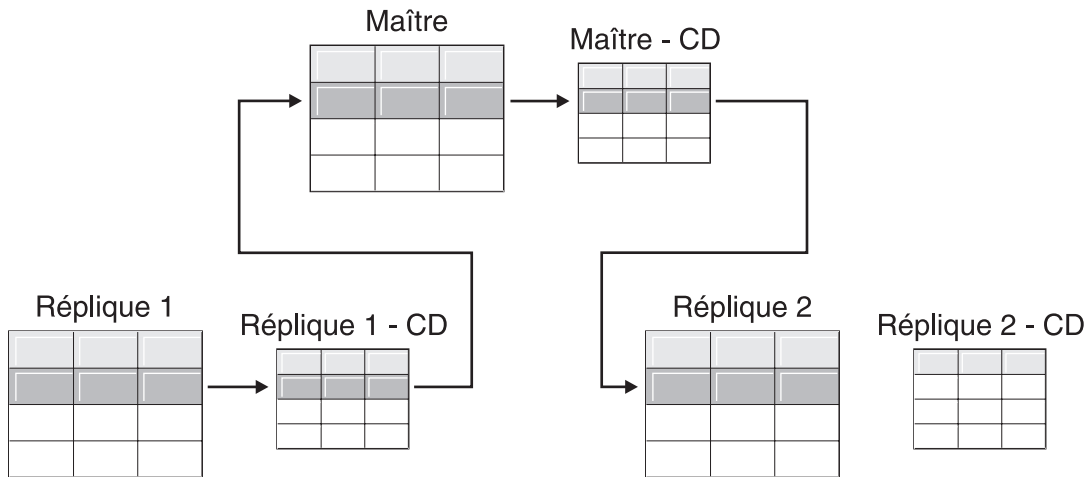


Figure 2. Option de recapture pour les tables maître qui répliquent les modifications dans plusieurs répliques. Lorsque plusieurs répliques sont abonnées aux mêmes données dans la table maître, vous pouvez utiliser l'option de recapture dans la table maître afin que les modifications apportées à une réplique soient recapturées dans la table maître et transférées vers les autres tables réplique.

## Répliques qui effectuent la réplcation des modifications dans d'autres répliques (multiniveau)

Pour les répliques qui répliquent les modifications dans plusieurs répliques (multiniveau), créez votre enregistrement de telle sorte que les modifications ne soient pas recapturées dans la table maître mais qu'elles le soient dans les tables réplique.

Vous devez posséder une configuration multiniveau dans laquelle la table maître (niveau 1) agit en tant que source de réplique (niveau 2), tandis que cette réplique est également la source d'une autre réplique (niveau 3). Si vous envisagez ce type de configuration, le programme Capture devra peut-être recapturer les modifications au milieu de la réplique (niveau 2), afin que les modifications apportées au niveau de la table maître soient transférées vers la réplique suivante (niveau 3).

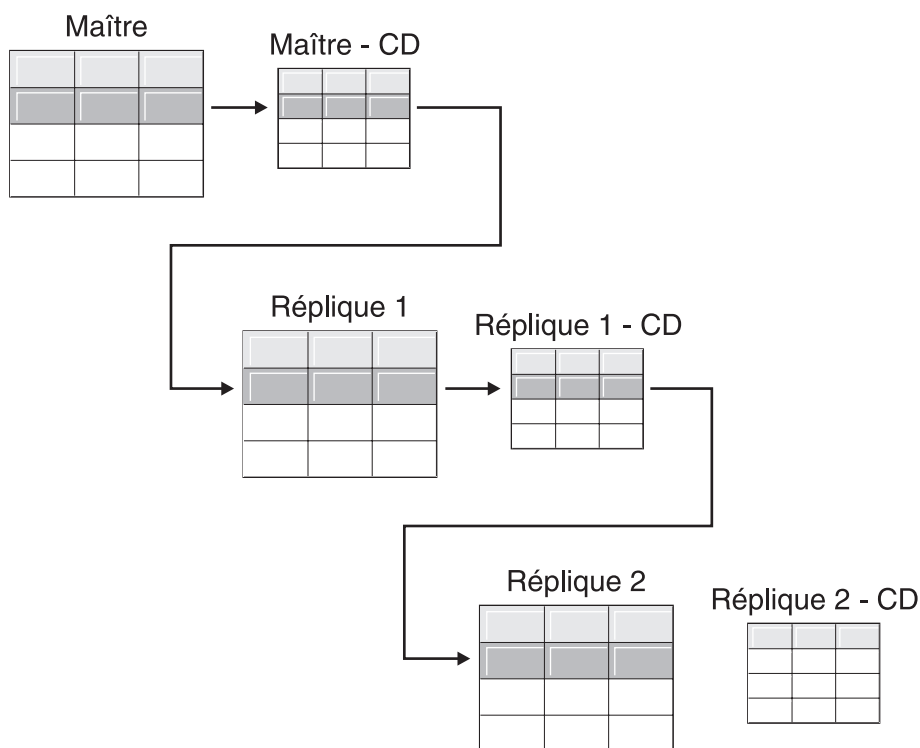


Figure 3. L'option de recapture (niveau 2) permet de répliquer les modifications de niveau 1 vers le niveau 3. Lorsqu'une table réplique est utilisée comme niveau intermédiaire au sein d'une configuration multiniveau, vous pouvez utiliser l'option de recapture au niveau de la réplique, afin que les modifications apportées à la table maître soient recapturées au niveau de la réplique (niveau intermédiaire), puis transférées vers la réplique de niveau suivant.

De la même façon, lorsque vous utilisez un ensemble de recapture pour la réplique de niveau intermédiaire (niveau 2), les modifications issues de la réplique finale (niveau 3) sont recapturées au niveau 2 et transférées vers la table maître (niveau 1).

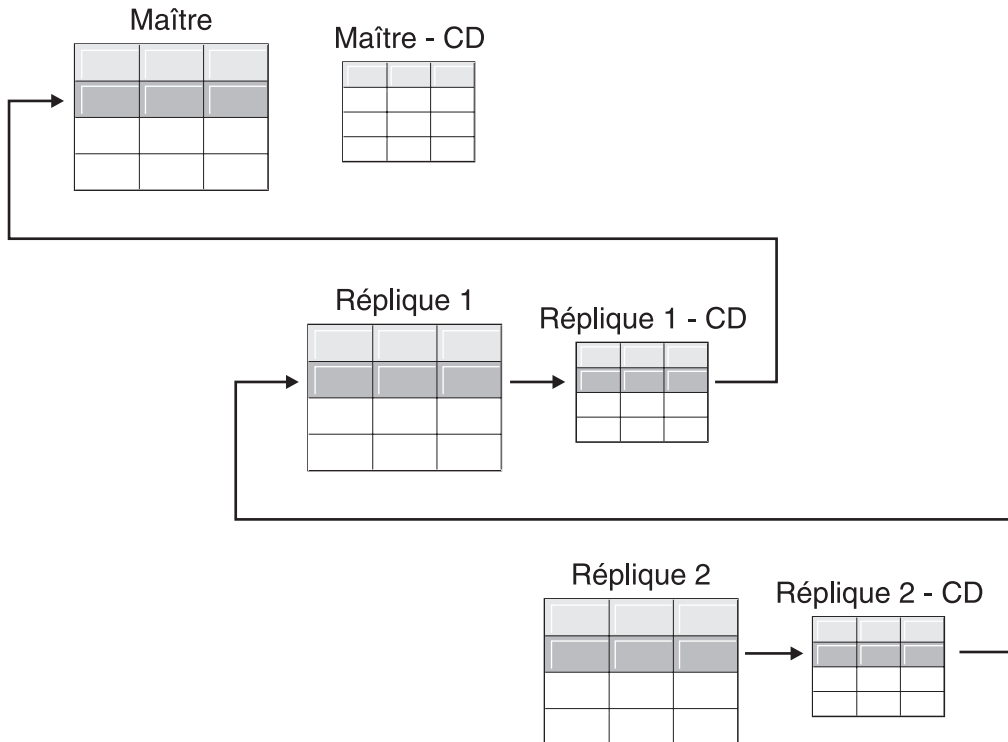


Figure 4. L'option de recapture (niveau 2) permet de répliquer les modifications de niveau 3 vers le niveau 1. Lorsqu'une table réplique est utilisée comme niveau intermédiaire au sein d'une configuration multiniveau, vous pouvez utiliser l'option de recapture au niveau de la réplique, afin que les modifications apportées à la réplique au niveau suivant soient recapturées au niveau de la réplique (niveau intermédiaire), puis transférées vers la réplique.

## Options de détection de conflit (réplication bidirectionnelle)

Dans les configurations bidirectionnelles, des conflits surviennent parfois entre la table maître et ses répliques. Lorsque vous enregistrez une source, vous pouvez sélectionner un niveau de détection de conflit parmi les trois niveaux suivants : aucun, standard et avancé.

Des conflits peuvent survenir dans les cas suivants :

- Lorsqu'une mise à jour est effectuée sur l'une des lignes de la table maître, qu'une autre mise à jour est effectuée sur la même ligne d'une ou de plusieurs tables réplique et que le programme Apply traite les modifications conflictuelles au cours d'un même cycle.
- Des contraintes sont violées

Même si vous définissez le niveau de détection de conflit applicable aux différentes sources de réplication, le programme Apply utilise le niveau le plus élevé de l'un des membres d'un ensemble d'abonnements comme niveau applicable à tous les autres membres de l'ensemble.

### Restrictions :

- Les tables issues de bases de données relationnelles non DB2 ne peuvent pas être utilisées pour les réplifications bidirectionnelles. Par conséquent, les sources relationnelles non DB2 ne disposent pas de détection de conflit.
- Si votre configuration de réplication bidirectionnelle inclut des colonnes LOB, vous devez spécifier la valeur Aucun en tant que niveau de détection de conflit.



En fonction de votre tolérance aux pertes ou aux transactions rejetées, et sur la base de vos exigences en matière de performances, vous pouvez déterminer le type de détection de conflit à utiliser :

#### **Aucune**

Pas de détection de conflit. Les mises à jour conflictuelles entre la table maître et la table réplique ne sont pas détectées. Cette option n'est pas recommandée pour la réplification bidirectionnelle.

#### **Standard**

Détection de conflit modérée.

Au cours de chaque cycle Apply, le programme Apply compare les valeurs clés figurant dans la table CD de la table maître et les valeurs incluses dans la table CD de la réplique. Si les mêmes valeurs clés figurent dans les deux tables CD, un conflit survient. En cas de conflit, le programme Apply annule la transaction qui avait été validée au niveau de la réplique via la lecture de la table CD de la réplique et la conservation des modifications issues de la table maître uniquement.

#### **Avancée**

Il s'agit du niveau de détection de conflit qui offre l'intégrité des données la plus élevée, tant au niveau de la table maître que de ses répliques.

Comme pour la détection standard, le programme Apply compare les valeurs clés figurant dans la table CD de la table maître et les valeurs incluses dans la table CD de la réplique au cours de chaque cycle Apply. Si les mêmes valeurs clés figurent dans les deux tables CD, un conflit survient. Toutefois, avec la détection avancée, le programme Apply attend que toutes les transactions en attente aient été validées pour effectuer une recherche de conflit. Pour vérifier l'interception de toutes les transactions, le programme Apply les recherche dans toutes les tables cible de l'ensemble d'abonnements et commence la détection de conflit une fois toutes les modifications capturées dans la table de modification des données (CD). En cas de conflit, le programme Apply annule la transaction qui avait été validée au niveau de la réplique via la lecture de la table CD de la réplique et la conservation des modifications issues de la table maître uniquement.

**Restriction** : même si vous spécifiez un niveau de détection de conflit avancé, lorsque le programme Apply est exécuté en environnement occasionnellement connecté (démarré via le mot clé COPYONCE), le programme Apply utilise la détection de conflit standard.

Le programme Apply ne peut pas détecter les dépendances de lecture. Par exemple, si une application lit des informations qui sont ensuite supprimées (via l'exécution d'une instruction DELETE ou d'une transaction d'annulation), le programme Apply n'est pas capable de détecter la dépendance.

Si vous définissez une configuration de réplification dans laquelle les conflits sont possibles (pour cela, sélectionnez la valeur Aucune ou Standard), vous devez inclure une méthode d'identification et de gestion des conflits. Même si l'infrastructure de réplification a détecté et sauvegardé vos mises à jour de transactions conflictuelles, le concepteur de l'application doit déterminer les opérations à entreprendre pour les transactions qui ont été validées et qui se trouvent maintenant annulées. La routine d'exit ASNDONE est exécutée en fin de cycle d'abonnement, c'est pourquoi le concepteur de l'application peut l'utiliser comme point de départ de cette logique spécifique. Les informations relatives aux

mises à jour conflictuelles annulées restent dans les tables CD ET UOW jusqu'à ce qu'elles deviennent admissibles pour l'élagage de durée de conservation.

## Enregistrement des tables qui utilisent la journalisation distante (System i)

Lorsque vous enregistrez des tables System i utilisant la consignation distante, vous pouvez définir le journal distant comme source de réplication au lieu du journal local.

En sélectionnant l'option de journalisation distante pour la réplication, vous déplacez les tables de données de modification, et les tables de contrôle Capture dans un serveur de base de données System i qui est distinct du serveur System i sur lequel se trouve la table source.

Lorsque vous enregistrez des tables sur System i en tant que sources, la valeur par défaut suppose que vous ne voulez pas utiliser la journalisation distante.

**Recommandation :** chaque fois que vous répliquez des données d'une table System i dans une autre table System i et qu'un journal distant est configuré, l'utilisation de la fonction de journalisation distante est très recommandée lors de l'enregistrement. En effet, l'utilisation de la journalisation distante dans la réplication améliore nettement les performances. Grâce à la fonction de journal distant, il est possible de déplacer l'enregistrement, le programme Capture ainsi que les tables de contrôle Capture hors du système sur lequel se trouve la table source, et ainsi, plus de ressources sont laissées à la disposition de ce système. Cela réduit l'utilisation du processeur et économise de l'espace disque. De même, lorsque vous utilisez un journal distant qui se trouve sur le serveur cible, la table de données de modification se trouve sur le même système que la table cible, ce qui permet au programme Apply d'appliquer les modifications directement depuis la table de données de modification dans la table cible, sans devoir utiliser le fichier auxiliaire. Le fait de ne pas utiliser de fichier auxiliaire réduit la quantité de ressources utilisées par le programme Apply.

**Recommandation :** n'enregistrez des tables qui utilisent les journaux distants en tant que source, que si l'enregistrement se trouve sur le même système System i que la cible de réplication. La réplication SQL vous permet d'enregistrer des journaux distants en tant que sources, même si l'enregistrement ne se trouve pas sur le même système System i que la cible. Cependant vous n'obtenez pas les avantages de performances qu'en ayant le journal sur le système cible.

Avant d'enregistrer une table System i qui utilise la journalisation distante, assurez-vous que votre journal distant est en état actif.

**Restrictions :** Les tables enregistrées qui utilisent la journalisation distante sont soumises aux restrictions suivantes :

- Les types de table réplique cible ne sont pas pris en charge dans une configuration de journal distant.
- L'option de requête `SQL_FAST_DELETE_ROW_COUNT` (ou suppression rapide) provoque l'arrêt de la journalisation. Elle ne doit pas être utilisée pour les tables enregistrées. Pour pallier cette restriction, vous pouvez utiliser une clause `WHERE` dans la commande de suppression ou définir le paramètre `SQL_FAST_DELETE_ROW_COUNT` de QAQQINI sur "none". La commande de suppression rapide n'a pas pour effet de consigner les suppressions une par une dans l'historique.

- Pour réorganiser la table source, n'utilisez pas l'option ALWCANCEL \*YES de la commande RGZPFM. L'utilisation de cette option entraîne la création d'une entrée de journal CE, ce qui conduit le programme Capture à signaler une régénération intégrale. Pour réorganiser une table source de réplication, indiquez ALWCANCEL \*NO dans la commande RGZPFM.

Pour plus d'informations sur la fonction de journal distant, voir "Remote journal management" dans le centre de documentation d'i5/OS.

## Intégrité référentielle sur la table cible lorsque la source est System i

Lorsque vous répliquez des données issues de tables source sur System i, vous pouvez rencontrer des contraintes d'intégrité référentielle (RI) sur les tables cible, mais certaines restrictions s'appliquent.

La réplication prend en charge les contraintes RI sur la cible lorsque la source se trouve sur System i et si les conditions suivantes sont respectées :

- Les contraintes RI sur les tables source et cible correspondent.
- L'application source utilise un contrôle de validation explicite.
- L'application source valide les modifications effectuées sur les tables enfant et parent. C'est-à-dire, la transaction parent est séparée de celle de ses enfants.
- Les tables parent et enfant sont consignées dans le même journal.
- Les tables parent et enfant se trouvent dans le même ensemble d'abonnements.
- La valeur de COMMIT\_COUNT dans la table IBMSNAP\_SUBS\_SET n'est pas nulle pour l'ensemble, ce qui force le programme Apply à traiter les changements dans un ordre transactionnel.
- Une régénération intégrale est possible en supprimant les contraintes RI sur le côté cible lorsque l'application est mise au repos. L'intégrité référentielle est ensuite réintégré une fois les tables synchronisées. Un changement de test est effectué et répliqué avant la publication de l'application.

D'autres facteurs peuvent avoir une influence sur la réplication, tels que l'existence d'une table parent ne faisant pas partie de la réplication. Par défaut, Le programme Apply traite les tables dans un ensemble d'abonnements par ordre de table et procède à une validation uniquement lorsque toutes les tables de l'ensemble ont été traitées.

Pour les cibles copie utilisateur, nous vous recommandons de disposer des contraintes RI sur la source et non sur la cible, et de placer toutes les tables concernées dans le même ensemble d'abonnements. Un ensemble est traité comme une unité unique, ainsi, en cas d'erreur de traitement d'une table, l'intégralité de l'ensemble échoue. Par conséquent, l'ensemble d'abonnements pourrait fournir des résultats acceptables.

## Utilisation de numéros relatifs d'enregistrement (RRN) à la place de clés primaires (System i)

Si vous enregistrez une table System i qui ne comporte pas de clé primaire, ou d'index à entrées uniques, ou de combinaison de colonnes pouvant être utilisées comme index à entrées uniques, vous devez enregistrer la table à l'aide des numéros relatifs d'enregistrement (RRN).

Lorsque vous choisissez de répliquer à l'aide de RRN, la table de données de modification et la table cible possèdent une colonne supplémentaire,

IBMQSQ\_RRN de type INTEGER (nombre entier), qui contient une valeur unique pour chaque ligne. Cette colonne contient le RRN qui correspond à chaque ligne de la table source.

Le RRN est utilisé comme clé primaire pour la ligne de table source tant que la table source n'est pas réorganisée. Lorsque la table source est réorganisée, le RRN de chaque ligne de la table source change ; c'est pourquoi le RRN dans les données de modification et les lignes de la table cible n'ont plus les valeurs exactes qui illustrent la nouvelle position de la ligne dans la table source.

Vous pouvez à tout moment réorganiser une table source (pour compresser les lignes supprimées, par exemple), DB2 DataPropagator pour System i effectue une régénération complète de toutes les tables cible dans l'ensemble de cette table source. C'est pourquoi vous devez placer des tables cible qui utilisent des RRN en tant que clés primaires dans des ensembles d'abonnements avec les autres cibles qui utilisent des RRN, et non dans des ensembles avec des tables qui utilisent un autre facteur d'unicité.

---

## Comportement des vues en tant que sources de réplication

Lorsque vous enregistrez des vues pour l'exécution de la réplication, ces vues héritent des options d'enregistrement de leurs tables sous-jacentes (et tout particulièrement de l'option de capture des modifications ou de l'option de régénération intégrale).

Les rubriques suivantes décrivent le comportement des vues dans le cadre de différents scénarios.

### Vues relatives à une seule table

Vous pouvez enregistrer une vue relative à une seule table si la table sous-jacente est enregistrée pour la réplication. La vue hérite du type de réplication de la table sous-jacente.

#### Régénération intégrale uniquement

Si la table sous-jacente est enregistrée pour une réplication avec régénération intégrale uniquement, la vue bénéficie de réplication avec régénération intégrale uniquement. Vous ne pouvez pas enregistrer la vue pour la réplication avec option de capture des modifications, car la table sous-jacente ne possède pas de table CD associée pour le suivi des modifications.

#### Capture des modifications

Si la table sous-jacente est enregistrée pour une réplication avec option de capture des modifications, la vue bénéficie de réplication avec option de capture des modifications et ne peut pas être enregistrée avec option de régénération intégrale uniquement.

Lorsque vous enregistrez une vue relative à une table enregistrée pour la réplication avec capture des modifications, une vue est créée pour la table CD de la table sous-jacente. Cette vue de modification des données (CD) contient uniquement les colonnes indiquées par la vue enregistrée.

Vous ne pouvez pas enregistrer un sous-ensemble de colonnes dans la vue. Toutes les colonnes de la vue sont enregistrées automatiquement.

## Vues relatives à la jonction de deux tables ou davantage

Lorsque vous enregistrez une vue via la jonction de deux tables ou davantage, l'une des tables sous-jacentes doit être enregistrée. Vous pouvez également effectuer la jonction interne de tables CCD qui ont été enregistrées en tant que sources.

Lorsque vous enregistrez une jonction en tant que source de réplication, la réplication SQL ajoute plusieurs lignes à la table IBMSNAP\_REGISTER portant des valeurs identiques pour SOURCE\_OWNER et pour SOURCE\_TABLE. Ces lignes diffèrent par leur valeur SOURCE\_VIEW\_QUAL. Chacune de ces entrées identifie un composant de la jonction.

**Restriction :** Si vous définissez une jonction qui inclut une table CCD, toutes les autres tables de la jonction doivent être des tables CCD.

Pour qu'une vue de jointure puisse être une source de réplication viable, vous devez la créer à l'aide d'un ID de corrélation. (Les vues de tables simples ne nécessitent pas d'ID de corrélation.)

### Exemple :

```
create view REGRES1.VW000 (c000,c1001,c2001,c2002,c1003) as
select a.c000,a.c001,b.c001,b.c002,a.c003
from REGRES1.SRC001 a, REGRES1.SRC005 b
où a.c000=b.c000;
```

VW000 représente le nom de la vue, SRC001 et SRC005 représentent les tables qui font partie de la vue et C000, C001, C002 et C003 représentent les colonnes qui font partie de la vue (sous réserve que les colonnes C000 soient équivalentes dans les deux tables (SRC001 et SRC005).

Le type de réplication dont hérite la vue dépend de la combinaison de ses tables sous-jacentes ; chacune d'entre elles peut être :

- Enregistrée pour la réplication en mode capture des modifications
- Enregistrée pour la réplication avec régénération intégrale uniquement
- Non enregistrée

Le tableau 3 indique les différentes combinaisons de tables sous-jacentes et le type de vue source et de vue de modification des données résulte de chaque combinaison.

Tableau 3. Combinaisons de tables sous-jacentes de vues

Tableau 1	Tableau 2	Description de la vue de jointure et de la vue de modification des données
Enregistrée pour la capture des modifications	Enregistrée pour la capture des modifications	La vue est enregistrée pour la réplication en mode capture des modifications. Les vues de modification des données contiennent les colonnes référencées issues de la table de modification des données du tableau 1 et du tableau 2.
Enregistrée pour la capture des modifications	Enregistrée pour la régénération intégrale uniquement	La vue est enregistrée pour la réplication en mode capture des modifications. La vue de modification des données contient les colonnes référencées issues de la table de modification des données du tableau 1 et du tableau 2. Seules les modifications apportées aux colonnes du tableau 1 sont répliquées dans la cible de la vue enregistrée au cours de chaque cycle de réplication.

Tableau 3. Combinaisons de tables sous-jacentes de vues (suite)

Tableau 1	Tableau 2	Description de la vue de jointure et de la vue de modification des données
Enregistrée pour la régénération intégrale uniquement	Enregistrée pour la régénération intégrale uniquement	La vue est enregistrée pour la réplication avec régénération intégrale uniquement. Il n'existe pas de vue de modification des données (CD).
Enregistrée pour la régénération intégrale uniquement	Non enregistrée	La vue est enregistrée pour la réplication avec régénération intégrale uniquement. Il n'existe pas de vue de modification des données (CD).
Enregistrée pour la capture des modifications	Non enregistrée	La vue est enregistrée pour la réplication en mode capture des modifications. La vue de modification des données contient les colonnes référencées issues de la table de modification des données du tableau 1 et du tableau 2. Seules les modifications apportées aux colonnes du tableau 1 sont répliquées dans la cible de la vue enregistrée au cours de chaque cycle de réplication.
Non enregistrée	Non enregistrée	La vue ne constitue pas une source de réplication valide et ne peut pas être enregistrée.

## Eviter les suppressions doubles

Lorsque vous définissez une vue qui inclut deux tables source (ou davantage) en tant que source de réplication, vous devez éviter les suppressions doubles. Une suppression double a lieu lorsqu'au cours du même cycle de réplication, vous supprimez une ligne dans les deux tables qui composent une vue. Par exemple, supposons que vous souhaitiez créer une vue contenant la table CUSTOMERS et la table CONTRACTS. Une suppression double se produit si vous supprimez une ligne de la table CUSTOMERS et que vous supprimez également la ligne correspondante (sous l'angle de la jointure de tables) de la table CONTRACTS au cours du même cycle de réplication. Le problème réside dans le fait que la ligne n'apparaît pas dans les vues, car elle a été supprimée des deux tables source (vue de base et vue de table de modification des données) ; par conséquent, la suppression double ne peut pas être répliquée dans la cible.

Pour éviter les suppressions doubles, vous devez définir une table CCD pour l'une des tables source de la jointure. Cette table CCD doit être condensée et incomplète. Elle doit par ailleurs se trouver sur le serveur cible. La définition d'une table CCD condensée et incomplète pour l'une des tables source de la jointure permet, dans la plupart des cas, de résoudre le problème des suppressions doubles : en effet, la colonne IBMSNAP\_OPERATION de la table CCD permet de détecter les suppressions. Il vous suffit donc d'ajouter une instruction SQL à la définition de l'ensemble d'abonnements devant être exécuté après le cycle d'abonnement. Cette instruction SQL supprime toutes les lignes de la table cible dont la valeur de IBMSNAP\_OPERATION est égale à la valeur «D» de la table CCD.

Des incidents de mise à jour et de suppression peuvent malgré tout se produire si, au cours d'un même cycle Apply, une ligne est mise à jour dans la table source possédant la table CCD, tandis que la ligne correspondante est supprimée dans l'autre table de la jointure. Lorsque cela se produit, le programme Apply ne trouve pas la ligne correspondante dans la table jointe et ne parvient pas à répliquer la valeur mise à jour.

---

## Enregistrement de vues de tables en tant que sources

Lorsque vous enregistrez une vue en tant que source de réplication, cette vue hérite des options d'enregistrement de la table source prise comme base.

### Avant de commencer

- Les tables de contrôle Capture doivent se trouver sur le serveur de contrôle Capture qui traitera la vue à enregistrer en tant que source.
- Le nom des vues source doivent se conformer à la convention de dénomination DB2 applicable aux tables.
- Vous devez enregistrer au minimum une des tables de base sous-jacentes de la vue en tant que source. Lorsque vous enregistrez la table de base, utilisez le schéma Capture que vous envisagez d'utiliser pour l'enregistrement de la vue.

### Restrictions

- Vous ne pouvez pas enregistrer les vues de tables relationnelles non DB2.
- Vous ne pouvez pas enregistrer une vue située sur une autre vue.
- Toutes les tables CCD dans lesquelles des vues ont été définies doivent être complètes et condensées pour pouvoir être enregistrées en tant que source de réplication.
- **System i** La longueur des instructions SQL est limitée à 32 000 caractères ; par conséquent, vous ne pouvez enregistrer qu'environ 2000 colonnes par vue ; le nombre exact de colonnes dépend de la longueur des noms de colonne.

### Procédure

Utilisez l'une des méthodes suivantes pour enregistrer une vue :

Méthode	Description
<b>Programme de ligne de commande ASNCLP</b>	Utilisez la commande CREATE REGISTRATION et spécifiez le nom de la vue pour <i>propriétaireobjet</i> et <i>nomobjet</i> .  Pour les vues, la commande détermine si la source peut être enregistrée en tant que régénération intégrale ou différentielle.
<b>Centre de réplication</b>	Utilisez la fenêtre Enregistrer les vues. Développez le schéma Capture à utiliser pour l'enregistrement de vues. Cliquez avec le bouton droit de la souris sur le dossier <b>Vues enregistrées</b> et cliquez sur <b>Enregistrer les vues</b> .
<b>System i</b> Commande système <b>ADDDPRREG</b>	Utilisez la commande ADDDPRREG pour enregistrer une vue sous System i.

---

## Gestion de tables de modification cohérente des données (CCD) en tant que sources

Si vous possédez des tables CCD externes gérées par un programme tel que IMS DataPropagator ou DataRefresher, vous devez gérer ces tables de telle sorte que le programme Apply puisse les lire comme des tables source.

### Procédure

Pour gérer une table CCD remplie à l'aide d'un outil externe, procédez comme suit :

Mettez à jour trois colonnes de la table IBMSNAP\_REGISTER (CCD\_OLD\_SYNCHPOINT, SYNCHPOINT et SYNCHTIME) pour chacun des types d'événements suivants :

Événement	Mises à jour requises
<b>Régénération intégrale initiale ou charge de la table CCD</b>	<ul style="list-style-type: none"> <li>• Affectez à CCD_OLD_SYNCHPOINT une valeur représentant la valeur minimale de IBMSNAP_COMMITSEQ, dans la table CCD.</li> <li>• Affectez à SYNCHPOINT une valeur représentant la valeur maximale de IBMSNAP_COMMITSEQ, dans la table CCD. N'affectez pas la valeur 0 à SYNCHPOINT. Si vous créez vos propres valeurs, commencez par la valeur 1 pour SYNCHPOINT.</li> <li>• Affectez à SYNCHTIME une valeur qui représente la valeur maximale d'horodatage de IBMSNAP_LOGMARKER, dans la table CCD.</li> </ul>
<b>Mises à jour de la table CCD effectuées après la régénération intégrale</b>	<ul style="list-style-type: none"> <li>• Ne modifiez pas la valeur de CCD_OLD_SYNCHPOINT.</li> <li>• Affectez à SYNCHPOINT une valeur représentant la nouvelle valeur maximale de IBMSNAP_COMMITSEQ, dans la table CCD.</li> <li>• Affectez à SYNCHTIME une valeur qui représente la nouvelle valeur maximale d'horodatage de IBMSNAP_LOGMARKER, dans la table CCD.</li> </ul>
<b>Régénération intégrale ultérieure ou charge de la table CCD</b>	<ul style="list-style-type: none"> <li>• Affectez à CCD_OLD_SYNCHPOINT une valeur représentant la valeur minimale de IBMSNAP_COMMITSEQ, dans la table CCD.</li> <li>• Affectez à SYNCHPOINT une valeur représentant la valeur maximale de IBMSNAP_COMMITSEQ, dans la table CCD.</li> <li>• Affectez à SYNCHTIME une valeur qui représente la valeur maximale d'horodatage de IBMSNAP_LOGMARKER, dans la table CCD.</li> </ul>

**Important :** Cela suppose que les valeurs utilisées dans la table CCD pour IBMSNAP\_COMMITSEQ et IBMSNAP\_LOGMARKER soient toujours des valeurs d'augmentation. Le programme Apply ne détecte pas qu'une régénération intégrale a été effectuée au niveau de la table CCD source, sauf si la valeur de CCD\_OLD\_SYNCHPOINT est plus élevée que la valeur de SYNCHPOINT dont l'application est la plus récente.



---

## Chapitre 5. Abonnement à des sources pour la réplication SQL

Une fois des tables ou des vues enregistrées comme sources de réplication, vous pouvez définir un abonnement pour vos tables et vues cible afin qu'elles reçoivent les données source initiales et les modifications ultérieures.

Les tâches d'administration décrites dans cette section permettent de configurer les informations de contrôle que les programmes Capture et Apply utilisent pour copier des données source ou capturer des données modifiées et les répliquer dans des tables cible à l'intervalle approprié.

Les rubriques suivantes fournissent des détails sur l'abonnement à des sources.

---

### Planification du mode de regroupement des sources et des cibles

Avant de définir quelles cibles s'abonnent à quelles sources, vous devez planifier comment regrouper vos sources et vos cibles.

La réplication SQL traite des mappages de sources à cibles sous forme de groupes. Ces groupes se composent d'une ou plusieurs sources traitées par le même programme Capture et d'une ou plusieurs cibles s'abonnant à tout ou partie des données source, lesquelles sont traitées par le même programme Apply. Ces groupes sont appelés *ensembles d'abonnements* et les mappages de sources à cibles sont des *membres d'un ensemble d'abonnements*.

Lorsque vous planifiez des ensembles d'abonnements, pensez aux règles et contraintes suivantes :

- Un ensemble d'abonnements mappe un serveur source avec un serveur cible. Un membre d'un ensemble d'abonnements mappe une table ou vue source avec une table ou vue cible. Les ensembles d'abonnements et leurs membres sont stockés dans le serveur de contrôle Apply.
- Le programme Apply traite tous les membres d'un ensemble d'abonnements comme un seul groupe. C'est pourquoi, si un membre d'un ensemble d'abonnements requiert une copie avec régénération intégrale, tous les membres de cet ensemble sont aussi régénérés.
- Toutes les tables et vues source dans les membres d'un ensemble doivent posséder le même schéma de Capture.
- Sur System i,, toutes les tables source dans les membres d'un ensemble d'abonnements doivent être consignées dans le même journal.
- Toutes les tables CCD externes créées par IMS DataPropagator et qui sont membres d'un ensemble d'abonnements doivent posséder le même schéma de Capture.

Un même programme Apply, avec un qualificatif Apply unique, peut traiter un ou plusieurs ensembles d'abonnements. Un même ensemble d'abonnements peut contenir un ou plusieurs membres.

Les rubriques suivantes présentent les compromis de regroupement d'ensembles d'abonnements par programme Apply et des membres par ensemble d'abonnements.

## Planification du nombre de membres d'ensembles d'abonnements

Lorsque vous ajoutez des membres à un ensemble d'abonnements, vous devez décider si vous voulez regrouper toutes les paires source/cible (membres) dans un même ensemble d'abonnements, créer des ensembles d'abonnements distincts pour chaque paire ou créer un petit nombre d'ensembles d'abonnements, chacun avec quelques paires.

Comme le programme Apply réplique les membres d'un ensemble d'abonnements lors d'une même transaction (logique), vous devez regrouper plusieurs membres dans un ensemble d'abonnements dans les cas suivants :

- si les tables source sont associées de façon logique entre elles,
- si les tables cible possèdent des contraintes d'intégrité référentielle.

En regroupant plusieurs membres dans un même ensemble d'abonnements, vous garantissez que la réplication commence à la même heure pour tous les membres. Par ailleurs, vous réduisez le nombre de connexions de bases de données requises pour traiter les ensembles d'abonnements et limitez le temps système d'administration pour la maintenance de l'environnement de réplication. Si l'ensemble d'abonnements contient des instructions SQL ou des procédures stockées, vous pouvez utiliser celles-ci pour traiter tous les membres de l'ensemble d'abonnements.

S'il n'existe aucune relation logique ou d'intégrité référentielle entre les tables d'un ensemble d'abonnements, vous pouvez les regrouper dans un ou plusieurs ensembles d'abonnements. La raison principale pour limiter le nombre d'ensembles d'abonnements est de simplifier l'administration de l'environnement de réplication. En augmentant le nombre d'ensembles d'abonnements, vous réduisez l'impact des échecs de réplication.

Pour pouvoir détecter plus facilement des erreurs faisant échouer le programme Apply, ajoutez quelques membres seulement à un ensemble d'abonnements. Avec peu de membres, vous avez des chances de trouver plus vite la cause de l'incident que si l'ensemble comporte un grand nombre de membres. Si un membre d'un ensemble d'abonnements échoue, toutes les données appliquées à d'autres membres de l'ensemble sont annulées et aucun membre ne peut achever le cycle si tous les autres ne le font pas. Le programme Apply annule un ensemble d'abonnements ayant échoué jusqu'à son dernier point de validation réussi, lequel peut se trouver dans le cycle actuel du programme Apply si vous avez entré le mot clé `commit_count` au démarrage du programme Apply.

## Planification du nombre d'ensembles d'abonnements par qualificatif Apply

Lorsque vous définissez un ensemble d'abonnements, vous indiquez le qualificatif Apply correspondant. Ce dernier associe alors une instance du programme Apply à un ou plusieurs ensembles d'abonnements.

Chaque ensemble d'abonnements est traité par un seul programme Apply, mais chaque programme Apply peut traiter un ou plusieurs ensembles d'abonnements lors d'un cycle du programme Apply.

Vous pouvez exécuter autant d'instances du programme Apply (chacune avec son propre qualificatif Apply) que nécessaire et chaque programme Apply peut traiter autant d'ensembles d'abonnements que requis. Vous disposez de deux options de base :

#### **Associer chaque qualificatif Apply à un ensemble d'abonnements.**

Chaque programme Apply traite alors un seul ensemble d'abonnements.

Si la vitesse compte, vous pouvez répartir les ensembles entre plusieurs qualificatifs Apply, ce qui vous permet d'exécuter simultanément plusieurs instances du programme Apply.

Si vous décidez qu'une instance du programme Apply doit traiter un ensemble d'abonnements, vous pouvez utiliser l'option de démarrage OPT4ONE chargeant en mémoire les informations des tables de contrôle pour l'ensemble d'abonnements.

Avec cette option, le programme Apply ne lit pas les tables de contrôle pour les informations de l'ensemble d'abonnements de chaque cycle du programme Apply. Par conséquent, le programme Apply fonctionne mieux. Toutefois, plus vous exécutez d'instances du programme Apply, plus elles utiliseront des ressources système et moins les performances globales seront élevées.

#### **Associer chaque qualificatif Apply à plusieurs ensembles d'abonnements**

Chaque programme Apply traite alors plusieurs ensembles d'abonnements.

Avec plusieurs qualificatifs Apply, vous pouvez exécuter plusieurs instances du programme Apply à partir d'un même ID utilisateur.

Le programme Apply tente de conserver aussi actualisés que possible tous les ensembles pour un qualificatif Apply donné. Au démarrage d'un cycle du programme Apply, le programme Apply identifie les ensembles d'abonnements contenant le moins de données en cours et lance d'abord le traitement de cet ensemble.

Si la vitesse n'est pas déterminante, vous pouvez répliquer un grand nombre d'ensembles d'abonnements avec un seul qualificatif Apply. Cette solution est par exemple idéale si vous répliquez après les heures ouvrables.

L'inconvénient qu'un seul programme Apply traite plusieurs ensembles d'abonnements est que le traitement est séquentiel et que le temps d'attente global de réplication peut augmenter.

Dans le cas de besoins spécifiques pour certains ensembles d'abonnements, vous pouvez combiner ces deux options. Par exemple, un programme Apply peut traiter la plupart des ensembles d'abonnements ayant un rapport entre eux et un autre un seul ensemble d'abonnements : de cette façon, le temps d'attente de réplication est minimal pour cet ensemble d'abonnements. Si vous utilisez deux instances du programme Apply, vous augmentez le parallélisme global de vos ensembles d'abonnements.

---

## **Création d'un ensemble d'abonnements**

Avant de répliquer des données à partir d'une source enregistrée, vous devez créer des ensembles d'abonnements, qui sont des regroupements de membres (mappages source-cible) que le programme Apply traite comme formant un ensemble.

### **Avant de commencer**

- Créez les tables de contrôle Apply sur le serveur de contrôle Apply pour l'ensemble d'abonnements concerné.
- Avant d'ajouter des membres aux ensembles d'abonnements, vous devez enregistrer les tables ou les vues que vous utiliserez comme sources. Vous devez également déterminer la façon dont vous souhaitez regrouper vos ensembles.

### A propos de cette tâche

Lorsque vous créez un ensemble d'abonnements, vous spécifiez les serveurs source et cible, les programmes Capture et Apply à utiliser, ainsi que le moment et la méthode de traitement de l'ensemble par le programme Apply.

Il est inutile d'ajouter des membres d'ensembles d'abonnements à un ensemble d'abonnements. Vous pouvez créer un ensemble vide, ne contenant aucun mappage source-cible. Vous aurez peut-être besoin de créer un ensemble vide pour les raisons suivantes :

- Vous envisagez d'ajouter ultérieurement des membres à un ensemble d'abonnements et ne souhaitez pas activer cet ensemble avant de lui avoir ajouté des membres.
- Vous souhaitez que le programme Apply traite l'ensemble d'abonnements vide afin d'appeler une instruction SQL ou une procédure stockée chaque fois que l'ensemble est admissible pour le traitement.

### Procédure

Pour créer un ensemble d'abonnements, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Programme de ligne de commande ASNCLP</b>	<p>Utilisez la commande CREATE SUBSCRIPTION SET. Cette commande permet uniquement de créer des ensembles d'abonnements vides, tandis que le Centre de réplication permet d'ajouter des membres à l'ensemble lors de sa création.</p> <p>Les commandes suivantes définissent l'environnement et créent un ensemble d'abonnements (SET00) portant le qualificatif Apply AQ00.</p> <pre>SET SERVER CAPTURE TO DB SAMPLE; SET SERVER CONTROL TO DB TARGET; SET OUTPUT CAPTURE SCRIPT "capsubset.sql" CONTROLSRIPT "appsubset.sql"; SET LOG "subset.err"; SET RUN SCRIPT LATER; CREATE SUBSCRIPTION SET SETNAME SET00 APPLYQUAL AQ00 ACTIVATE YES TIMING INTERVAL 1 START DATE "2006-10-22" TIME "09:00:00.000000";</pre>
<b>Centre de réplication</b>	<p>Utilisez le bloc-notes Créer un ensemble d'abonnements. Pour l'ouvrir, développez le serveur de contrôle Apply sur lequel l'ensemble doit être défini, puis cliquez avec le bouton droit de la souris sur le dossier <b>Ensembles d'abonnements</b> ; ensuite, cliquez sur <b>Créer</b>.</p>

Méthode	Description
<p style="text-align: center;"><b>System i</b></p> <p>Commande système <b>ADDDPRSUB</b></p>	<p>Utilisez la commande d'ajout d'ensemble d'abonnements DPR pour créer un ensemble d'abonnements avec membres ou sans membre.</p> <p>Par exemple, pour créer un ensemble d'abonnements (SETHR) sous le qualificatif Apply AQHR, procédez comme suit :</p> <pre>ADDDPRSUB APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/EMPLOYEE) TGTTBL(TGTLIB/TGTEMPL)</pre> <p>Cet ensemble d'abonnements, qui contient un membre, réplique les données de la table source enregistrée EMPLOYEE de la bibliothèque HR dans la table cible TGTEMPL de la bibliothèque TGTLIB.</p>

Vous devez spécifier les caractéristiques de base suivantes :

#### **Alias du serveur de contrôle Apply**

Alias local du serveur contenant les tables de contrôle pour le programme Apply qui sera chargé de traiter l'ensemble d'abonnements. Définissez le même alias pour le serveur de contrôle Apply dans chaque base de données utilisée pour l'exécution du Centre de réplication, du programme ASNCLP ou du programme Apply : cela permet aux outils d'administration de remplir correctement les tables de contrôle Apply et à chaque programme Apply de se connecter au serveur approprié à l'aide d'un nom d'alias standard.

#### **Nom d'ensemble d'abonnements**

Nom de l'ensemble d'abonnements. Sur le serveur de contrôle Apply qui traite cet ensemble d'abonnements, le nom d'ensemble doit être unique pour un qualificatif Apply donné. Ce nom peut comporter au maximum 18 caractères.

#### **Qualificatif Apply**

Nom d'un qualificatif Apply (nouveau ou existant), qui mentionne le programme Apply chargé du traitement de cet ensemble d'abonnements. Vous pouvez utiliser le même qualificatif Apply pour traiter plusieurs ensembles d'abonnements. Les ensembles d'abonnements qui portent le même qualificatif Apply doivent être définis sur le même serveur de contrôle Apply.

#### **Alias du serveur de contrôle Capture**

Alias du serveur contenant les tables de contrôle pour le programme Capture qui sera chargé de traiter les sources enregistrées de l'ensemble d'abonnements. Définissez le même alias pour le serveur de contrôle Capture dans chaque base de données utilisée pour l'exécution du Centre de réplication, du programme ASNCLP ou du programme Apply : cela permet aux outils d'administration de remplir correctement les tables de contrôle Apply et Capture, et à chaque programme Apply de se connecter au serveur approprié à l'aide d'un nom d'alias standard.

#### **Schéma Capture**

Nom du schéma Capture identifiant l'ensemble de tables de contrôle Capture qui définit les sources enregistrées de l'ensemble d'abonnements. Toutes les tables source d'un ensemble d'abonnements doivent résider sur le même serveur, et un seul programme Capture peut effectuer la capture des modifications pour ces tables.

### Alias du serveur cible

Nom du serveur cible contenant les tables ou les vues dans lesquelles le programme Apply répliquera les modifications provenant de la source. Définissez le même alias pour le serveur cible dans chaque base de données utilisée pour l'exécution du Centre de réplication, du programme ASNCLP ou du programme Apply : cela permet aux outils d'administration de remplir correctement les tables de contrôle Apply et à chaque programme Apply de se connecter au serveur approprié à l'aide d'un nom d'alias standard.

Lorsque vous ajoutez un ensemble d'abonnements, vous pouvez utiliser les valeurs par défaut pour indiquer au programme Apply le traitement souhaité pour l'ensemble ; vous pouvez également modifier les propriétés d'abonnement pour les adapter à vos besoins en matière de réplication.

---

## Options de traitement pour les ensembles d'abonnements

Lorsque vous créez un ensemble d'abonnements, vous définissez des options pour son mode de traitement par le programme Apply.

Les rubriques suivantes permettent de décider quels paramètres choisir en fonction de vos besoins de réplication.

### Indication si l'ensemble d'abonnements est actif

Vous pouvez indiquer si le programme Apply doit lancer le traitement de l'ensemble d'abonnements. Lorsque vous activez un ensemble d'abonnements, le programme Apply entame une régénération intégrale de celui-ci.

Il existe trois niveaux d'activation possibles :

**Actif** Le programme Apply traite l'ensemble au cours du cycle suivant. Activez l'ensemble si vous voulez que le programme Apply le traite à sa prochaine exécution. Vous pouvez toujours ajouter plus tard des membres à l'ensemble. Lorsque vous activez l'ensemble, il reste actif et le programme Apply en poursuit le traitement tant que vous ne le désactivez pas.

**Inactif** Le programme Apply ne traite pas l'ensemble. Laissez-le inactif si vous ne voulez pas que le programme Apply le traite.

#### Actif une seule fois

Le programme Apply traite l'ensemble au cours du cycle suivant, puis le désactive. Indiquez cette option pour que l'ensemble ne soit exécuté qu'une seule fois. Assurez-vous d'ajouter tous les membres de l'ensemble d'abonnements avant de sélectionner cette option, car le programme Apply ne traitera pas les membres ajoutés par la suite, sauf si vous réactivez l'ensemble d'abonnements.

### Indication du nombre de minutes de données extraites par le programme Apply

Vous pouvez indiquer un nombre approximatif de minutes de données que le programme Apply doit extraire de la source de réplication lors de chaque cycle du programme Apply.

Cette option est utile dans diverses situations :

- Lorsque la quantité de données à traiter dans un cycle d'ensemble d'abonnements est élevée.

Les ensembles d'abonnements répliquant des blocs longs de modifications dans un cycle du programme Apply peuvent provoquer le dépassement des journaux ou des fichiers auxiliaires (pour la base de données cible). Par exemple, les scénarios par lots du programme Apply peuvent générer de nombreuses commandes en attente de transactions ajoutées à la file d'attente et devant être répliquées.

- Une indisponibilité durable du réseau peut entraîner l'accumulation d'un bloc long de données dans les tables CD, d'où un dépassement du fichier auxiliaire du programme Apply et du journal de la cible.

Le nombre de minutes indiqué est qualifié de bloc de données. La valeur de groupage de réplication indiquée est stockée dans la colonne MAX\_SYNCH\_MINUTES de la table IBMSNAP\_SUBS\_SET. Si l'accumulation des données dépasse la taille du bloc de données, le programme Apply convertit un cycle du programme Apply en plusieurs mini-cycles. Si les ressources demeurent insuffisante pour gérer le facteur de groupage fourni, le programme Apply réduit la taille du bloc de données pour qu'elle corresponde aux ressources système disponibles. En extrayant des ensembles de données plus petits, le programme Apply peut alléger la charge du réseau et l'espace temporaire requis pour les données extraites.

Lors de chaque cycle du programme Apply, si la valeur MAX\_SYNCH\_MINUTES d'un ensemble d'abonnements est NULL ou est une valeur numérique inférieure à 1, le programme Apply traite toutes les données admissibles pour chaque ensemble dans un même cycle. Si les tables CD et UOW contiennent d'importants volumes de données, des incidents peuvent se produire : remplissage du journal des transactions de la base de données, dépassement du fichier auxiliaire. Vous pouvez changer la valeur MAX\_SYNCH\_MINUTES en prenant une valeur non NULL en suivant ces instructions :

- Si la colonne SLEEP\_MINUTES de la table ASN.IBMSNAP\_SUBS\_SET affiche 5 minutes (ou moins) pour un ensemble d'abonnements donné, entrez 5 minutes pour MAX\_SYNCH\_MINUTES.
- Si SLEEP\_MINUTES affiche 30 minutes (ou plus) pour un ensemble d'abonnements donné, entrez 60 minutes pour MAX\_SYNCH\_MINUTES.
- Si la valeur dans la colonne SLEEP\_MINUTES est comprise entre 5 et 30 minutes, entrez la même valeur pour MAX\_SYNCH\_MINUTES.

Surveillez votre environnement de réplication et modifiez la valeur MAX\_SYNCH\_MINUTES en conséquence. Vérifiez que la valeur numérique pour MAX\_SYNCH\_MINUTES est supérieure à zéro.

**Exemple** : si vous indiquez que le programme Apply doit extraire au moins 10 minutes de données par mini-cycle, il extraira une quantité de données validées de la table CD à la source figurant à environ 10 minutes du dernier mini-cycle.

En plus d'empêcher le dépassement des journaux et des fichiers auxiliaires, ces mini-cycles présentent divers avantages. Si une erreur se produit lors du cycle de réplication, le programme Apply doit uniquement annuler les modifications effectuées lors du mini-cycle ayant échoué. Si la réplication échoue au cours d'un mini-cycle, le programme Apply tente de traiter l'ensemble d'abonnements depuis le dernier mini-cycle ayant abouti, ce qui peut permettre de gagner beaucoup de temps si une quantité importante de données est disponible pour traitement.

La figure 5 affiche la manière dont les données modifiées sont réparties en sous-ensembles de modifications.

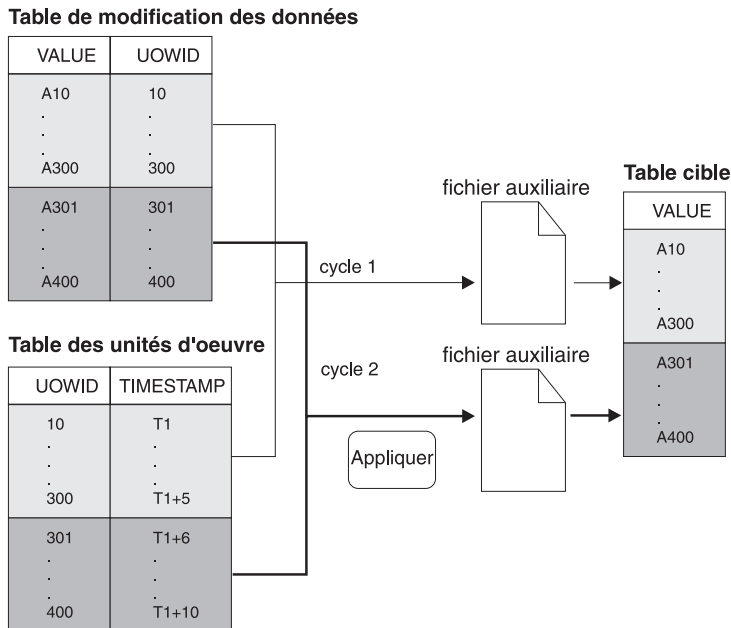


Figure 5. Groupage de réplication. Vous pouvez réduire la quantité de trafic réseau en indiquant une valeur de groupage de réplication.

Le nombre de minutes défini doit être suffisamment bas pour que toutes les transactions pour l'ensemble d'abonnements se produisant lors de l'intervalle puissent être copiées sans entraîner le dépassement des journaux et des fichiers auxiliaires lors du mini-cycle.

Lors du traitement des données, le programme Apply n'effectue aucune des actions suivantes :

- Division d'une unité de travail (un travail par lots de longue durée ne peut pas être divisé par le facteur de groupage de réplication).
- Annulation de mini-cycles d'abonnement validés auparavant.
- Utilisation du facteur de groupage de réplication lors d'une régénération intégrale.

## Options de chargement pour les tables cible avec intégrité référentielle

Dans certains cas, vous souhaitez reporter les contraintes d'intégrité référentielle entre tables cible afin qu'elles interviennent après le chargement des tables avec les données source.

Vous définissez le chargement des cibles au moment de la configuration des paramètres de démarrage pour le programme Apply. Pour la création de relations d'intégrité référentielle entre les tables cible, vous avez le choix entre les possibilités suivantes :

### Création avant le chargement des tables cible

Cela implique qu'aucune modification ne soit apportée à la table source pendant la totalité de l'opération d'extraction et de chargement de la table cible. Vous devez par ailleurs démarrer le programme Apply à l'aide de



l'option LOADX, afin d'ignorer la vérification de contraintes référentielles pendant le chargement. Si vous n'utilisez pas l'option LOADX, les insertions dans la table cible sont susceptibles d'échouer. Une régénération intégrale est généralement plus rapide lorsque vous utilisez cette option.

### **Création une fois le chargement terminé et une fois que le programme Apply a effectué un cycle complet d'application des modifications aux cibles**

Les modifications peuvent dans ce cas être apportées à la table source, tandis que les tables cible sont chargées. Vous pouvez démarrer le programme Apply avec ou sans l'option LOADX, car il n'y a pas de contraintes à ignorer. Au cours du remplissage initial des tables cible, les cibles peuvent avoir des relations d'intégrité référentielle non synchronisées. Au fur et à mesure du chargement des tables, toutes les modifications sont capturées pour l'ensemble. Une fois que le programme Apply a répliqué le premier ensemble de modifications, toutes les tables cibles contiennent les mêmes transactions et présentent une intégrité référentielle. A ce stade, vous pouvez désactiver l'ensemble, ajouter les contraintes d'intégrité référentielle, puis réactiver l'ensemble.

## **Spécification de la réplication des modifications pour les membres d'ensembles d'abonnements par le programme Apply**

Lorsqu'un ensemble d'abonnements fait l'objet d'une réplication en mode capture de modifications, vous pouvez déterminer si le programme Apply doit valider les modifications dans la table ou la vue cible pour chaque membre ou après avoir appliqué un certain nombre de transactions.

Après le chargement initial des tables cible, le programme Apply commence à lire les tables de modification des données (CD) ou les tables de modification cohérente des données (CCD) et collecte les modifications dans des fichiers auxiliaires. Ensuite, le programme applique les modifications de l'une des deux façons suivantes :

### **Mode table**

Le programme Apply valide les modifications pour chaque membre d'ensemble d'abonnements.

Le programme Apply lit toutes les modifications dans un fichier auxiliaire, pour les tables de modification des données (CD) et de modification cohérente des données (CCD) ; ensuite, il applique les modifications aux tables cible correspondantes, puis commence à traiter le fichier auxiliaire pour la table CD ou CCD suivante. Une fois qu'il a terminé la lecture et l'application de toutes les modifications des tables CD et CCD de l'ensemble, il émet une validation DB2 afin de valider toutes les modifications dans toutes les tables cible de l'ensemble d'abonnements.

### **Mode transaction**

Le programme Apply valide les modifications après avoir appliqué le nombre de transactions spécifié. Utilisez le mode transaction lorsque des contraintes d'intégrité référentielle existent sur les tables cible d'un ensemble d'abonnements.

En mode transaction, le programme Apply ouvre tous les fichiers auxiliaires et traite les modifications en même temps. Les modifications sont appliquées dans l'ordre dans lequel elles ont été apportées dans les tables source. La colonne COMMIT\_COUNT de la table

IBMSNAP\_SUBS\_SET contrôle l'application et la validation des modifications dans toutes les tables cible de cet ensemble d'abonnements.

Le mode transaction change le comportement du programme Apply uniquement pour les ensembles possédant des tables cible de type CCD, copie utilisateur et point de cohérence. Les ensembles contenant des tables réplique sont toujours traités en mode transaction.

Une validation peut diminuer le temps d'attente de l'ensemble d'abonnements, mais plusieurs validations permettent au programme Apply d'appliquer les données dans l'ordre de validation d'origine.

Vous pouvez également utiliser une combinaison de mode table et de mode transaction, en fonction du type de table cible associé à l'ensemble d'abonnements.

## Définition d'instructions SQL ou de procédures stockées pour l'ensemble d'abonnements

Vous pouvez définir des instructions SQL ou des procédures stockées s'exécutant chaque fois que le programme Apply traite l'ensemble d'abonnements. Ces instructions peuvent être utiles pour la suppression de tables CCD ou la manipulation de données source avant leur application à des cibles.

Vous pouvez indiquer quand et où les instructions SQL ou les procédures stockées doivent s'exécuter :

- sur le serveur de contrôle de Capture avant l'application des données par le programme Apply,
- sur le serveur cible avant l'application des données par le programme Apply,
- sur le serveur cible après l'application des données par le programme Apply.

Lorsque vous recourez au Centre de réplication pour ajouter des instructions SQL à un ensemble d'abonnements, vous pouvez cliquer sur **Préparation de l'instruction** dans la fenêtre Ajout d'une instruction SQL ou de l'appel de procédure afin de vérifier la syntaxe.

## Options de planification de la réplication d'ensembles d'abonnements

Vous pouvez indiquer la fréquence à laquelle le programme Apply traite un ensemble d'abonnements afin de contrôler le niveau d'actualité des données dans vos tables cible. Vous pouvez recourir à une planification fondée sur l'horaire et/ou un événement.

Vous pouvez par exemple définir un intervalle d'un jour entre les cycles du programme Apply et indiquer un événement déclenchant un cycle. Si vous employez les deux options de planification, il est possible de traiter l'ensemble d'abonnements tant à l'heure planifiée qu'au moment de l'événement.

Dans une réplication bidirectionnelle, vous pouvez utiliser une planification identique ou distincte pour les ensembles d'abonnements de maître à réplique et de réplique à maître.

Dans le cas d'une quantité importante de données à répliquer au cours d'un intervalle ou entre des événements, le programme Apply peut ne pas traiter un ensemble d'abonnements tant qu'il n'a pas appliqué les données pour tous les ensembles dans l'intervalle antérieur ou lors de l'événement précédent. Dans ce

cas, vous ne bénéficiez éventuellement pas du temps d'attente de réplication souhaité, mais vous ne perdez aucune donnée.

## Planification fondée sur l'horaire

La méthode la plus simple pour contrôler à quel moment un ensemble est traité consiste à employer une planification fondée sur l'horaire (également dite à intervalles réguliers ou synchronisation temporelle). Vous fixez dans ce cas une date et une heure de début, ainsi qu'un intervalle. Ce dernier peut être spécifique (d'une minute à une année) ou continu, mais les intervalles de temps sont approximatifs.

Le programme Apply lance dès que possible le traitement d'un ensemble d'abonnements, en fonction de sa charge de travail et de la disponibilité des ressources. Le choix d'un intervalle de temps ne garantit pas que la fréquence de réplication correspondra exactement. Si vous optez pour une planification continue, le programme Apply réplique les données aussi souvent que possible.

## Planification fondée sur un événement

Pour répliquer des données à l'aide d'une planification fondée sur un événement (également qualifiée de synchronisation événementielle), vous indiquez un nom d'événement au moment de définir l'ensemble d'abonnements. Vous devez aussi renseigner la table IBMSNAP\_SUBS\_EVENT avec un horodatage pour le nom de l'événement. Lorsque le programme Apply détecte l'événement, il entame la réplication.

La table IBMSNAP\_SUBS\_EVENT possède quatre colonnes, comme illustré dans le tableau 4.

Tableau 4. Exemple de données stockées dans la table IBMSNAP\_SUBS\_EVENT

EVENT_NAME	EVENT_TIME	END_OF_PERIOD	END_SYNCHPOINT
END_OF_DAY	2002-05-01- 17.00.00.000000	2002-05-01- 15.00.00.000000	

La colonne EVENT\_NAME stocke le nom de l'événement indiqué lors de la définition de l'ensemble d'abonnements. EVENT\_TIME est l'horodatage de début de traitement de l'ensemble par le programme Apply. END\_OF\_PERIOD est une valeur facultative indiquant que les mises à jour qui se produisent après l'heure indiquée doivent être différées jusqu'à un événement ou une heure ultérieurs. END\_SYNCHPOINT est également une valeur facultative indiquant que les mises à jour qui se produisent après le numéro de séquence de journal doivent être différées jusqu'à un événement ou une heure ultérieurs. Si vous entrez des valeurs pour END\_OF\_PERIOD et END\_SYNCHPOINT, la valeur de END\_SYNCHPOINT l'emporte. Définissez la valeur EVENT\_TIME à l'aide de l'horloge du serveur de contrôle Apply et la valeur END\_OF\_PERIOD à l'aide de l'horloge du serveur source. Cette distinction est importante si les deux serveurs se trouvent dans des zones horaires différentes.

Dans le tableau 4, pour l'événement END\_OF\_DAY, la valeur d'horodatage pour EVENT\_TIME (2002-05-01-17.00.00.000000) correspond à l'heure à laquelle le programme Apply doit commencer le traitement de l'ensemble d'abonnements. La valeur d'horodatage END\_OF\_PERIOD (2000-05-01-15.00.00.000000) correspond quant à elle à l'heure après laquelle les mises à jour ne sont pas répliquées (elles le

seront alors lors du cycle du lendemain). Dans ce cas, l'événement réplique toutes les mises à jour en attente effectuées avant cette heure et diffère toutes les suivantes.

Vous-même ou vos applications devez afficher des événements dans la table IBMSNAP\_SUBS\_EVENT à l'aide d'une instruction SQL INSERT afin d'insérer une ligne dans la table et d'activer l'événement. Par exemple, utilisez l'horodatage en cours plus une minute pour déclencher l'événement EVENT\_NAME. Tout ensemble d'abonnements associé à cet événement peut alors être exécuté en une minute. Vous devez afficher manuellement des événements tant pour la régénération intégrale que pour la répllication en mode capture des modifications.

Vous pouvez afficher des événements à l'avance (par exemple, pour la semaine ou l'année suivante, pour tous les samedis). Si le programme Apply est en cours d'exécution, il démarre environ à l'heure indiquée. S'il est arrêté à l'heure précisée, il vérifie la table d'événements d'abonnement et lance le traitement de l'ensemble d'abonnements pour l'événement affiché.

Le programme Apply ne supprime pas la table. Vous devez renseigner et gérer cette dernière. Par ailleurs, vous ne pouvez pas utiliser le Centre de répllication pour mettre à jour la table d'événements d'abonnement. Vous devez émettre des instructions SQL ou définir des procédures automatisées afin d'ajouter des événements à cette table.

**Exemple :**

```
INSERT
INTO ASN.IBMSNAP_SUBS_EVENT
    (EVENT_NAME, EVENT_TIME)
VALUES ('EVENT01', CURRENT_TIMESTAMP + 1 MINUTES)
```

Tout événement se produisant avant la dernière heure de traitement de l'ensemble d'abonnements par le programme Apply (comme indiqué par la valeur dans la colonne LASTRUN de la table de contrôle de l'ensemble d'abonnements) est considéré comme arrivé à expiration et donc ignoré. Par conséquent, si le programme Apply est en cours d'exécution, vous devez afficher des événements légèrement ultérieurs afin de ne pas afficher un événement arrivé à expiration.

## Planification de l'ensemble d'abonnements

Définissez les informations de planification de l'ensemble d'abonnements après avoir mappé les sources vers les cibles (ou créé un ensemble d'abonnements vide).

Une fois les sources mappées vers les cibles (ou un ensemble d'abonnements vide créé), définissez les informations de planification de l'ensemble d'abonnements. Dans la page Planification de la fenêtre Création d'un ensemble d'abonnements, indiquez à quel moment l'ensemble d'abonnements doit être disponible pour traitement ; la valeur par défaut est la date et l'heure en cours de la machine locale. Indiquez aussi la planification pour la fréquence à laquelle l'ensemble d'abonnements doit être disponible pour traitement :

- Réplication fondée sur l'horaire  
Le programme Apply traite cet ensemble d'abonnements avec un intervalle de temps régulier.
- Réplication fondée sur un événement  
Le valider traite cet ensemble d'abonnements chaque fois qu'un événement se produit.
- Réplication fondée sur l'horaire et sur un événement

Le programme Apply traite cet ensemble d'abonnements avec un intervalle de temps régulier et chaque fois qu'un événement se produit. Dans ce cas, l'ensemble d'abonnements peut être traité à la fois à l'heure planifiée et lorsque l'événement se produit.

## Création de membres d'ensembles d'abonnements

Au sein d'un ensemble d'abonnements, vous pouvez ajouter des mappages entre source et cible pour que le programme Apply effectue un traitement en tant que groupe. Ces mappages entre source et cible sont appelés membres d'ensembles d'abonnements.

### Avant de commencer

Avant de configurer des cibles qui s'abonneront aux modifications apportées aux sources, vous devez enregistrer les tables ou les vues à utiliser comme sources. Vous devez également créer un ensemble d'abonnements et déterminer le nombre de membres à ajouter à cet ensemble.

### Restrictions

- La réplication SQL ne prend pas en charge en tant que sources les vues de tables relationnelles non DB2.
- Si vous définissez une vue cible, cette vue doit pouvoir être insérée. Cela signifie que toutes les colonnes de la vue doivent pouvoir être mises à jour et que la sélection de la vue complète ne doit pas inclure les mots clés UNION ALL.
- Si vous utilisez le Centre de réplication, vous ne pouvez pas ajouter de colonne à un membre d'ensemble d'abonnements si cette colonne ne figure pas déjà dans la table cible.
- **z/OS** : Ne sélectionnez pas la colonne ROWID pour réplication à moins que celle-ci ne soit le seul index à entrées uniques indiqué pour la réplication.

**Recommandation** : Utilisez plutôt une colonne IDENTITY comme index à entrées uniques pour la réplication.

- **z/OS** **Linux UNIX Windows** Vous pouvez définir au maximum 200 membres par ensemble d'abonnements.
- **System i** Vous pouvez définir au maximum 78 membres par ensemble d'abonnements.

### A propos de cette tâche

Lorsque vous définissez un membre d'ensemble d'abonnements, vous indiquez la table cible ou la vue abonnée aux données source, et vous pouvez définir la forme sous laquelle les données répliquées doivent figurer dans la cible.

### Procédure

Pour ajouter un membre d'ensemble d'abonnements, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	<p>Utilisez la commande CREATE MEMBER pour ajouter un membre d'ensemble d'abonnements à un ensemble existant. Par exemple, les commandes indiquées ci-après permettent de :</p> <ul style="list-style-type: none"> <li>• définir l'environnement,</li> <li>• créer un profil (TBSPROFILE) pour le stockage des options destinées à l'espace table utilisé par la table cible,</li> <li>• spécifier l'ensemble d'abonnements SET00, le qualificatif Apply AQ00 et la table source STAFF,</li> <li>• indiquer qu'une nouvelle table cible (TRGSTAFF) est créée en tant que copie utilisateur avec enregistrement de toutes les colonnes.</li> </ul> <pre>SET SERVER CAPTURE TO DB SAMPLE; SET SERVER CONTROL TO DB TARGET; SET SERVER TARGET TO DB TARGET; SET OUTPUT CAPTURE SCRIPT "capmember.sql" CONTROLSCRIPT "appmember.sql" SET LOG "member.err"; SET RUN SCRIPT LATER; SET PROFILE TBSPROFILE FOR OBJECT TARGET TABLESPACE OPTIONS UW USING FILE "/tmp/db/ts/TSTRG.TS" SIZE 700 PAGES; CREATE MEMBER IN SETNAME SET00 APPLYQUAL AQ00 ACTIVATE YES SOURCE STAFF TARGET NAME TRGSTAFF DEFINITION IN TSTRG00 CREATE USING PROFILE TBSPROFILE TYPE USERCOPY COLSALL REGISTERED;</pre>
Centre de réplication	<p>Utilisez l'un des bloc-notes suivants :</p> <ul style="list-style-type: none"> <li>• Créer un ensemble d'abonnements. Utilisez ce bloc-notes lorsque vous créez l'ensemble d'abonnements. Développez le serveur de contrôle Apply sur lequel l'ensemble sera défini, puis cliquez avec le bouton droit de la souris sur le dossier <b>Ensembles d'abonnements</b> ; ensuite, cliquez sur <b>Créer</b>.</li> <li>• Propriétés de l'ensemble d'abonnements. Utilisez ce bloc-notes si vous avez déjà créé l'ensemble d'abonnements et que vous souhaitez lui ajouter un ou plusieurs membres. Cliquez avec le bouton droit de la souris sur l'ensemble d'abonnements, puis sélectionnez <b>Propriétés</b>.</li> <li>• Ajouter des membres aux ensembles d'abonnements. Utilisez ce bloc-notes pour ajouter un membre à plusieurs ensembles d'abonnements. Chaque membre doit utiliser la même source. Cliquez avec le bouton droit de la souris sur les ensembles d'abonnements auxquels vous souhaitez ajouter un membre, puis sélectionnez <b>Ajouter un membre</b>.</li> </ul>
<div style="background-color: #800000; color: white; padding: 2px; text-align: center; font-weight: bold;">System i</div> Commande système <b>ADDDPRSUBM</b>	<p>Utilisez la commande <b>ADDDPRSUBM</b> pour ajouter un membre à un ensemble d'abonnements. Par exemple, pour ajouter un membre d'ensemble d'abonnements à l'ensemble d'abonnements SETHR sous le qualificatif Apply AQHR, procédez comme suit :</p> <pre>ADDDPRSUBM APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/YTD TAX) TGTTBL(TGTHR/TGTTAX)</pre>

Pour effectuer le mappage entre une source et une cible, spécifiez les informations suivantes sur la table enregistrée ou sur la vue à utiliser comme source :

- Table ou vue source et table ou vue cible (incluant un espace table et un index pour la table cible).
- Type de table cible.
- Colonnes enregistrées de la table source à répliquer dans la table cible.

Lorsque vous utilisez le Centre de réplication pour mapper une source à une cible, les colonnes de données LOB ne sont pas automatiquement incluses dans le mappage de colonnes. Vous devez sélectionner explicitement ces colonnes.

- Lignes de la table source à répliquer dans la table cible (en incluant une clause WHERE pour spécifier les lignes).

**Pour effectuer le mappage entre la source choisie et une cible DB2, procédez comme suit :**

Spécifiez les informations suivantes sur la table ou la vue cible :

- Schéma.
- Nom de la table ou de la vue à utiliser comme cible.

**Valeur par défaut :** le nom par défaut provient du profil d'objet cible défini pour le serveur cible, s'il en existe un. Si vous n'avez pas défini ce profil, la valeur par défaut est TG, suivie du nom de la table ou de la vue source. (Par exemple, si le nom de votre table source est EMPLOYEE, le nom de votre table cible sera par défaut TGEMPLOYEE.)

- Type de table cible.

**Valeur par défaut :** copie utilisateur

Si la table cible spécifiée n'existe pas, les outils d'administration ou la commande système **ADDDPRSUBM** la créent.

**Pour effectuer le mappage entre la source choisie et une cible relationnelle non DB2, procédez comme suit :**

Spécifiez les informations suivantes sur la table cible :

- Schéma de pseudonyme.
- Pseudonyme.
- Schéma distant.
- Nom de la table distante.

**Valeur par défaut :** le nom par défaut provient du profil d'objet cible défini pour le serveur cible, s'il en existe un. Si vous n'avez pas défini ce profil, la valeur par défaut est TG, suivie du nom de la table ou de la vue source. (Par exemple, si le nom de votre table source est EMPLOYEE, le nom de votre table cible sera par défaut TGEMPLOYEE.)

- Type de table cible.

**Valeur par défaut :** copie utilisateur

Lorsque vous ajoutez un membre d'ensemble d'abonnements, vous pouvez utiliser le type de table cible par défaut de la copie utilisateur ; vous pouvez également sélectionner un autre type de table cible correspondant à vos besoins en matière de réplication.

Lorsque vous ajoutez un membre d'ensemble d'abonnements pour une table cible qui n'existe pas encore, vous pouvez utiliser les valeurs par défaut ; vous pouvez également modifier les propriétés du membre pour les adapter à vos besoins en matière de réplication. Vous pouvez tout d'abord extraire le type de table cible à utiliser, puis définir les propriétés de réplication des données dans cette cible par le programme Apply.

## Types de table cible

Le type de table cible dépend du mode d'affichage souhaité pour vos données et de la configuration de réplication. Vous pouvez utiliser une table existante comme cible ou en créer une.

## Restrictions

- Les attributs null des colonnes cible image-après doivent être compatibles avec ceux des colonnes de la table ou vue source. Employez l'expression SQL COALESCE pour permettre la compatibilité avec des colonnes existantes.
- Pour les tables source dans des bases de données relationnelles non DB2, vous pouvez uniquement définir les types suivants de tables cible :
  - Tables de copie utilisateur
  - Tables des points de cohérence
  - Tables CCD externes
- Les noms de toutes les tables cible relationnelles non DB2 et les index doivent respecter les conventions de dénomination de tables et d'index de DB2.
- **System i** Pour les tables source sur System i utilisant des colonnes RRN comme colonnes clés, vous pouvez uniquement définir les types suivants de tables cible :
  - Tables des points de cohérence
  - Tables CCD externes
- **z/OS** Pour les tables source dans un sous-système z/OS, l'algorithme de codage pour les tables CD et UOW doit être identique si le programme Apply est censé les joindre en vue de respecter une clause WHERE de l'ensemble d'abonnements pour une table de copie utilisateur.

## Types cible

Vous pouvez faire un choix parmi les types suivants de tables cible :

### Copie utilisateur

Table cible en lecture seule incluant uniquement les colonnes définies dans le membre d'un ensemble d'abonnements. Une table de copie utilisateur peut posséder la même structure que la table source ou inclure un sous-ensemble de colonnes source, avec ou sans colonnes image-avant ou calculées. La réplication SQL suppose qu'il s'agit de la seule application écrivant dans des tables cible de copie utilisateur. Les modifications directes dans les tables de copie utilisateur, apportées par des utilisateurs finaux ou des applications, peuvent être écrasées par la réplication SQL et entraîner une non correspondance des données dans les tables source et cible. Si vous devez mettre à jour les tables source et cible, envisagez une réplication bidirectionnelle.

### Point de cohérence

Table cible en lecture seule incluant les colonnes définies dans le membre d'un ensemble d'abonnements et une colonne d'horodatage. Une table des points de cohérence peut posséder la même structure que la table source ou inclure un sous-ensemble de colonnes source, avec ou sans colonnes image-avant ou calculées.

### Agrégation de base

Table cible en lecture seule utilisant des fonctions de colonne SQL (comme SUM et AVG) pour calculer des récapitulatifs du contenu entier de la table source.

Une table d'agrégation de base récapitule le contenu d'une table source. Elle peut aussi comporter un horodatage et indiquer quand le programme Apply a réalisé l'agrégation. Utilisez ce type de table pour effectuer le suivi régulier de l'état d'une table source.



### **Agrégation des modifications**

Table cible en lecture seule utilisant des fonctions de colonne SQL (comme SUM et AVG) pour calculer des récapitulatifs du contenu entier de modifications récentes dans la table source et stockées dans la table CD ou dans une CCD interne.

Une table d'agrégation des modifications récapitule le contenu d'une table CD ou d'une table CCD interne au lieu d'une table source. Elle inclut également deux horodatages pour marquer l'intervalle de temps au cours duquel les modifications ont été capturées (écrites dans la table CD ou CCD). Utilisez ce type de table pour effectuer le suivi des modifications (opérations UPDATE, INSERT et DELETE) réalisées entre des cycles de réplication.

### **CCD (consistent-change data)**

Table cible en lecture seule avec des colonnes supplémentaires pour les informations de contrôle de réplication. Ces colonnes incluent un numéro d'enregistrement de journal, un indicateur de modification de la table source avec une instruction SQL INSERT, DELETE ou UPDATE, ainsi qu'un numéro d'enregistrement de journal et un horodatage de l'instruction de validation associée à l'insertion, à la suppression ou à la mise à jour. Vous pouvez aussi inclure éventuellement des colonnes image-avant et des colonnes issues de la table UOW.

### **Réplique**

Table cible en lecture/écriture pour la réplication bidirectionnelle. Une table réplique est le seul type de table cible que vos programmes et les utilisateurs peuvent directement mettre à jour. Par conséquent, une table réplique reçoit des modifications de la table maître et des programmes et utilisateurs locaux. Elle peut posséder la même structure que la table source ou inclure un sous-ensemble de colonnes source, mais elle ne comporte pas d'autres colonnes de contrôle de réplication (comme des horodatages). Les tables réplique sont uniquement prises en charge pour des bases de données DB2.

Les rubriques suivantes décrivent les utilisations de chaque type cible et la façon de définir les propriétés des tables cible pour répondre à vos besoins de réplication :

### **Tables cible en lecture seule**

En fonction du mode d'affichage souhaité sur la cible pour les données source, vous pouvez faire en sorte que des tables cible en lecture seule contiennent une copie de la table ou vue source, un historique des modifications ou un récapitulatif.

Les rubriques suivantes offrent plus de détails sur ces types de cibles en lecture seule.

### **Cibles de copie utilisateur et de point de cohérence :**

Par défaut, une table de copie utilisateur est créée comme type cible lorsque vous définissez un membre d'un ensemble d'abonnements. Sélectionnez le point de cohérence comme type cible pour savoir à quelle heure les modifications ont été appliquées à la cible.

### **Copie utilisateur**

Utilisez le type par défaut si vous voulez que la table cible corresponde à la table source au moment où la copie est réalisée. Les tables de copie

utilisateur ne contiennent pas d'autres colonnes de contrôle de réplication mais peuvent comporter un sous-ensemble de lignes ou de colonnes dans la table source, ainsi que d'autres colonnes non répliquées.

#### **Point de cohérence**

Sélectionnez le point de cohérence comme type cible pour savoir à quelle heure les modifications ont été appliquées à la cible. Une cible de point de cohérence contient les mêmes données que la table source, avec une colonne d'horodatage en plus pour indiquer à quel moment le programme Apply a validé chaque ligne de la cible. La colonne d'horodatage est null à l'origine. Les tables des points de cohérence peuvent comporter un sous-ensemble de lignes ou de colonnes dans la table source ou d'autres colonnes non répliquées.

**Restriction :** DB2 empêche l'insertion de valeurs dans les colonnes d'une table DB2 définies comme AS IDENTITY GENERATED ALWAYS. Pour éviter cette restriction, vous pouvez :

- créer la table cible sans la clause IDENTITY,
- créer la table cible avec la colonne AS IDENTITY GENERATED BY DEFAULT.

#### **Cibles de type agrégat de base ou agrégat de modification :**

Vous pouvez créer des tables cible qui contiennent des récapitulatifs du contenu des tables source ou des modifications les plus récentes apportées aux données de tables source.

Pour les types de table cible d'agrégats, vous pouvez définir de nouvelles colonnes via l'utilisation des fonctions d'agrégat telles que COUNT, SUM, MIN, MAX ou AVG. Ces colonnes ne contiennent pas les données source d'origine ; elles contiennent les valeurs calculées de la fonction SQL définie. Le programme Apply ne crée pas les agrégations pendant la régénération intégrale : les lignes sont ajoutées petit à petit, lors du traitement de l'ensemble par le programme Apply. L'un des avantages de l'utilisation d'une table d'agrégat réside dans le fait que la réplication SQL peut répliquer des informations récapitulatives uniquement, plutôt que chaque ligne individuelle, ce qui permet d'économiser de la largeur de bande réseau et de l'espace dans la table cible.

#### **Cibles de base de type agrégat**

Utilisez une table cible de type agrégat pour suivre l'état d'une table source pendant un cycle de réplication. Pour chaque table cible d'agrégat, le programme Apply effectue l'agrégation (lit et calcule) à partir de la table source. Ce type de table inclut également un horodatage indiquant l'heure à laquelle le programme Apply a effectué l'agrégation.

Si une table source enregistrée ne possède qu'une table d'agrégat comme cible, il est inutile de capturer les modifications pour la table source.

**Exemple :** supposons que vous souhaitiez connaître le nombre moyen de vos clients par semaine. Si votre table source possède une ligne pour chaque client, le programme Apply peut calculer le nombre total de lignes de votre table source sur une base hebdomadaire, et enregistrer les résultats dans une table d'agrégat. Si vous effectuez l'agrégation chaque semaine, la table cible contiendra 52 entrées indiquant le nombre de clients pour chaque semaine de l'année.

## Cibles d'agrégats de modification

Utilisez une table cible d'agrégats de modification pour suivre les modifications (opérations UPDATE, INSERT et DELETE) apportées entre les cycles de réplication à la table source. Pour chaque table cible d'agrégats de modification, le programme Apply effectue l'agrégation (lit et calcule) à partir de la table CD ou CCD interne. Une table d'agrégats de modification inclut également deux horodatages pour indiquer l'intervalle de temps applicable aux insertions, par le programme Capture, des modifications dans la table CD ou CCD.

**Exemple :** supposons que vous souhaitiez connaître le nombre de vos clients supplémentaires par semaine (opérations INSERT) et de vos clients en moins par semaine (opérations DELETE). Vous pouvez compter alors le nombre de lignes insérées et de lignes supprimées dans la table CD sur une base hebdomadaire et enregistrez le chiffre obtenu dans une table d'agrégats de modification.

**Important :** Si la table source d'un membre d'ensemble d'abonnements est enregistrée pour une réplication avec régénération intégrale uniquement, vous ne pouvez pas utiliser une table cible d'agrégats de modification, qui nécessite une table CD ou CCD au niveau de la source.

### Cibles CCD :

Les tables cible CCD (Consistent-Change-Data) fournissent les données de transactions validées qui peuvent être lues et utilisées par d'autres applications, par exemple InfoSphere DataStage. Vous pouvez également utiliser une table de modification cohérente des données (CCD) pour effectuer un audit des données source ou conserver un historique sur l'utilisation des données.

Par exemple, vous pouvez suivre les comparaisons avant et après des données, le moment où des modifications sont apportées ainsi que l'ID utilisateur ayant apporté la modification dans la table source.

Pour définir une table cible en lecture seule qui conserve un historique de la table source, définissez la table CCD cible de telle sorte qu'elle possède les attributs suivants :

#### Non condensée

Pour conserver un enregistrement de toutes les modifications source, définissez la table CCD de telle sorte qu'elle soit non condensée, c'est-à-dire qu'elle possède une ligne par modification apportée. Les tables non condensées contiennent de nombreuses lignes portant la même valeur clé ; par conséquent, ne définissez pas un index unique. Une table CCD non condensée contient une ligne par opération UPDATE, INSERT ou DELETE, ce qui permet de constituer un historique des opérations effectuées dans la table source. Si vous capturez des opérations UPDATE en tant qu'opérations INSERT et DELETE (pour le partitionnement de colonnes), la table CCD contiendra deux lignes par mise à jour (une ligne pour l'opération DELETE et une ligne pour l'opération INSERT).

#### Complète ou incomplète

Vous pouvez déterminer si vous souhaitez que la table CCD soit complète ou incomplète. Les tables CCD incomplètes ne contiennent pas à l'origine d'ensemble complet de lignes source. Vous devez donc créer une table CCD incomplète pour pouvoir conserver un historique des mises à jour apportées à une table source (mises à jour effectuées depuis que le programme Apply a commencé à remplir la table CCD).

### Inclure des colonnes UOW

Pour les fonctions d'audit avancées, incluez les colonnes supplémentaires de la table UOW. Si vous avez besoin d'une identification davantage orientée utilisateur, des colonnes pour ID de corrélation et ID d'autorisation primaire pour DB2 pour z/OS, ou encore pour nom de travail et profil utilisateur System i sont disponibles dans la table UOW.

**Important pour la version 10.1 sous Linux, UNIX et Windows :** Si la base de données source est DB2 10.1 sous Linux, UNIX et Windows avec des noeuds multiples DB2 pureScale, les cibles CCD sont prises en charge seulement si les programmes Capture et Apply ont été migrés vers la version 10.1 et si le paramètre de **compatibility** Capture est défini sur 1001. Avec un seul noeud DB2 pureScale et **compatibility** défini sur 0801, les cibles CCD sont prises en charge même si la version du programme Apply est antérieure à 10.1.

Par définition, une table CCD comprend toujours les colonnes suivantes en plus des colonnes répliquées à partir de la table source :

Colonne	Description
IBMSNAP_INTENTSEQ	<b>Type de données :</b> VARCHAR(16) FOR BIT DATA ; <b>Null admis :</b> non  Un numéro de séquence qui identifie uniquement un changement. Cette valeur est croissante dans une transaction.  <b>z/OS</b> Numéro de séquence du journal (LRSN ou RBA) de chaque opération de mise à jour, de suppression et d'insertion.
IBMSNAP_OPERATION	<b>Type de données :</b> CHAR(1); <b>Null admis :</b> Non  Indicateur qui spécifie le type d'opération : I (INSERT), U (UPDATE) ou D (DELETE).
IBMSNAP_COMMITSEQ	<b>Type de données :</b> VARCHAR(16) FOR BIT DATA ; <b>Null admis :</b> non  Numéro de séquence pour chaque ligne au sein d'une transaction.  <b>z/OS</b> Numéro de séquence du journal (LRSN ou RBA) de l'enregistrement de validation source.
IBMSNAP_LOGMARKER	<b>Type de données :</b> TIMESTAMP; <b>Null admis :</b> Non  L'heure approximative à laquelle les données ont été validées.

Lorsque vous créez une table CCD incomplète (COMPLETE=N) à l'aide du programme de ligne de commande ASNCPLP ou du Centre de réplication, vous pouvez spécifier des colonnes de contrôle facultatives. Le tableau suivant décrit ces colonnes :

Colonne	Description
IBMSNAP_AUTHID	<p><b>Type de données</b> : VARCHAR(30); <b>Null admis</b> : Oui</p> <p>ID utilisateur qui a mis à jour la table source.</p> <p><b>z/OS</b> Cette colonne correspond à l'ID utilisateur primaire.</p>
IBMSNAP_AUTHTKN	<p><b>Type de données</b> : VARCHAR(30); <b>Null admis</b> : Oui</p> <p>Clé d'accès qui est associée à la transaction.</p> <p><b>z/OS</b> ID de corrélation (normalement un nom de travail) qui a exécuté la mise à jour source.</p>
IBMSNAP_PLANID	<p><b>Type de données</b> : VARCHAR(8); <b>Null admis</b> : Oui</p> <p><b>z/OS</b> Nom de plan associé à la transaction. Cette colonne doit être NULL pour DB2 for Linux, UNIX et Windows.</p>
IBMSNAP_UOWID	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui</p> <p>Identificateur de l'unité de travail (UOW) provenant de l'enregistrement de journal pour une ligne.</p> <p><b>z/OS</b> Identificateur de l'unité de travail, parfois appelé ID unité de récupération (URID) de la transaction.</p>

#### Tables cible CCD internes :

Si des modifications sont fréquemment apportées à une table source, vous pouvez créer une table CCD interne afin de faire un récapitulatif des modifications validées apportées à la source depuis le dernier cycle Apply.

La table de modification des données (CD) connaît des flux constants lorsque le programme Capture ajoute des modifications en provenance du journal ; par conséquent, la mémoire cache locale contenant les modifications source au sein de la table CCD constitue une source plus stable pour vos cibles.

Lorsque la table source d'origine est mise à jour, le programme Capture lit les modifications fréquentes dans le journal de la source et les insère dans la table CD de la source. Dans cette table CD, un programme Apply lit les modifications et remplit la table CCD interne. Vous pouvez configurer la table CCD interne de façon à ce qu'elle ne contienne que les modifications les plus récentes de chaque ligne de la table CD apportées au cours du dernier cycle. Par conséquent, la table CCD est statique entre les différents cycles Apply (le programme Apply effectue alors la réplication de la table CD dans la table CCD), ce qui en fait une source plus stable pour les cibles. Si vous condensez les modifications à partir de la source, vous pouvez accroître les performances globales en évitant de répliquer dans la table cible un grand nombre de mises à jour apportées à la même ligne.

Le programme Capture ajoute en permanence de nouvelles modifications à la table CD ; par conséquent, un deuxième programme Capture lit les modifications dans la table CCD interne et non dans la table CD, afin de ne pas répliquer différentes modifications dans différentes cibles. Cela permet de conserver la synchronisation entre les cibles. Le deuxième programme Apply utilise la table source d'origine pour les régénérations intégrales et la table CCD interne pour la réplification avec option de capture des modifications.

**Important pour la réplification bidirectionnelle :** Si vous définissez une table CCD interne, le programme Apply l'ignore lors du traitement d'un ensemble d'abonnements possédant une réplique comme cible ; il applique les modifications à la réplique à partir de la table CD de la source maître.

### Recommandations

- Définissez un membre d'ensemble d'abonnements entre la table source et la table CCD interne avant de définir d'autres membres d'ensembles d'abonnements entre la table source et d'autres tables cible. Cela permet au programme Apply d'utiliser la table CCD interne et non la table CD pour la réplification des modifications de la table source. Si vous définissez d'autres membres d'ensembles d'abonnements et que vous commencez la réplification en utilisant ces membres avant de définir la table CCD interne applicable à la table source, vous devrez peut-être effectuer une régénération intégrale de toutes les cibles de la table source.
- Combinez toutes les tables CCD internes au sein d'un ensemble d'abonnements afin de vous assurer que toutes les tables cible de la base de données source sont synchronisées.
- Même lorsque vous souhaitez que seul un sous-ensemble des colonnes source fréquemment modifiées soit appliqué à d'autres cibles, utilisez la valeur par défaut afin que toutes les colonnes source enregistrées soient répliquées dans la table CCD interne. De cette façon, vous pouvez utiliser la table CCD interne en tant que source pour les tables cible futures, qui auront peut-être besoin des données des autres colonnes enregistrées dans la table source d'origine. Pour les tables cible futures, seules les colonnes de la table CCD interne seront disponibles pour la réplification avec capture des modifications.

### Attributs des tables CCD internes

Les tables CCD internes sont utilisées en tant que sources implicites de réplification : vous ne pouvez pas définir explicitement ce type de table comme source de réplification. Lorsque vous ajoutez un membre d'ensemble d'abonnements, vous effectuez un mappage entre la table source d'origine (et non la table CCD interne) et la table cible. Les tables CCD internes portent les attributs suivants :

#### Interne

La table CCD agit en tant qu'alternative à la table CD source. Les informations relatives à la table CCD interne sont stockées sur la même ligne que dans la table source, dans la table IBMSNAP\_REGISTER. Les tables CCD internes ne possèdent pas leur propre ligne dans la table de registres. Le programme Apply réplique automatiquement les modifications d'une table CCD interne, le cas échéant, plutôt que celles d'une table CD. Il ne peut y avoir qu'une seule table CCD interne par source de réplification.

**Restriction :** La table utilisateur ne contient pas de colonnes calculées ; par conséquent, veillez à ne pas en inclure dans les abonnements CCD.

**Locale** La table CCD se trouve dans la même base de données que la table source.

### **Incomplète**

Le programme Apply utilise la table source d'origine pour les régénérations intégrales et non la table CCD interne (la table CDD est incomplète car la cible possédera une copie initiale de toutes les lignes source).

### **Condensée**

La table CCD interne est condensée, ce qui signifie qu'elle contient une ligne par valeur clé ; le programme Apply applique donc la modification la plus récente à chaque ligne de la table CCD, au lieu d'appliquer une ligne par modification.

### **Absence de colonnes UOW**

Les tables CCD internes ne prennent pas en charge les colonnes de table UOW supplémentaires. Vous ne pouvez pas utiliser de table CCD interne si vous avez déjà défini une table CCD cible qui contient des colonnes UOW.

## **Définition du niveau intermédiaire dans une configuration multi-niveaux**

Le modèle de réplication de base compte deux niveaux, avec une seule source et une ou plusieurs cibles. Vous pouvez également établir des configurations avec trois niveaux ou plus.

### **Restrictions**

La plateforme intermédiaire dans les configurations multiniveaux doit être une table DB2.

### **A propos de cette tâche**

Une configuration multi-niveaux possède une table source et une table cible, laquelle sert de source pour d'autres tables cible.

Une raison pour configurer un environnement de réplication multi-niveaux est de fournir des temps système stables pour les cibles à trois niveaux. Vous pouvez aussi éviter de nombreuses connexions de bases de données à votre système source, ce qui attribue le coût de connexion au deuxième niveau. Comme vous collectez des modifications du niveau 1 dans des tables CCD au niveau 2, vous pouvez contrôler la fréquence de réplication des modifications à chaque niveau et réduire le nombre de modifications répliquées sur la cible (niveau 3).

Dans un modèle à trois niveaux par exemple, le premier niveau (1) est la base de données source et le deuxième (2) à la fois la cible du niveau 1 et une source pour un troisième niveau de cibles (3). Le niveau 2 peut donc distribuer des modifications à une ou plusieurs bases de données à trois niveaux. Si votre configuration de réplication comporte plus de deux niveaux, ceux intermédiaires, qui servent de sources et de cibles, sont des tables CCD.

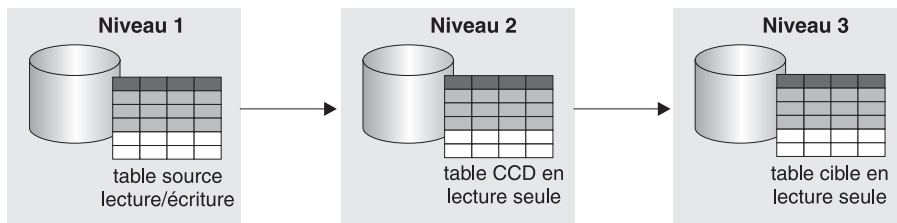


Figure 6. Modèle de réplication à trois niveaux. Vous pouvez répliquer des données d'une table source à une table cible, puis de cette table à une autre table cible.

Cette procédure s'applique aussi à des tables réplique. Les tables CCD sont généralement employées pour la réplication en lecture seule, alors que les tables réplique servent pour une réplication bidirectionnelle.

### Procédure

ou configurer une réplication multi-niveaux de façon à ce que votre table cible serve de source pour d'autres cibles :

1. Enregistrez la table source (niveau 1) pour la réplication. Le programme Capture pour cette source capture des modifications se produisant au niveau 1 et les stocke dans la table CCD de ce niveau.
2. Créez un ensemble d'abonnements entre le serveur source et le serveur cible (pour le niveau 2). Le programme Apply pour cet ensemble d'abonnements applique les modifications du niveau 1 à la table CCD au niveau 2.
3. Définissez le membre d'un ensemble d'abonnements mappant la table source (niveau 1) et une table cible CCD (niveau 2).

Lorsque vous définissez la table cible pour ce membre, faites en sorte que la table cible soit une table CCD avec les attributs suivants :

#### Source enregistrée externe

Vous devez définir la source comme table cible externe et enregistrer la table afin qu'elle agisse comme source pour le niveau suivant. Comme d'autres sources enregistrées, une table CCD externe possède sa propre ligne dans la table IBMSNAP\_REGISTER. Les tables CCD externes servant aussi de sources peuvent uniquement être remplies par une table source.

Vous devez enregistrer toutes les tables CCD externes dans un ensemble d'abonnements à l'aide du même schéma de Capture.

Vous pouvez répliquer dans une table CDD externe sans joindre la table des données de modification (CD) et la table IBMSNAP\_UOW. La nouvelle table est spécifiée avec une valeur de 9 dans la colonne TARGET\_STRUCTURE de la table IBMSNAP\_SUBS\_MEMBR. Bien que la table CCD de type 9 comprenne une colonne IBMSNAP\_LOGMARKER, le programme Apply n'exige pas de joindre la table CD et la table IBMSNAP\_UOW pour obtenir un horodatage de validation de source pour cette colonne. A la place, le programme Apply génère la même valeur dans la colonne IBMSNAP\_LOGMARKER pour toutes les lignes dans le même cycle. Le nouveau type de table CCD a la même structure que la table CDD de type 3. La table contient quatre colonnes IBM® obligatoires en plus des colonnes utilisateurs :



IBMSNAP\_COMMITSEQ  
IBMSNAP\_INTENTSEQ  
IBMSNAP\_OPERATION  
IBMSNAP\_LOGMARKER  
*colonnes\_utilisateur*

Ce type de table cible peut être enregistré comme table source pour une configuration de réplication à trois niveaux.

**Important :** Pour les tables CDD de type 9, le facteur de groupage des données (MAX\_SYNCH\_MINUTES dans la table de contrôle IBMSNAP\_SUBS\_SET) doit être désactivé (NULL).

### Complète

Vous devez utiliser une table CCD complète car le programme Apply s'en servira pour une régénération intégrale et une réplication en mode capture des modifications pour le niveau suivant.

### Condensée

Utilisez une table CCD condensée, à savoir qui contient une ligne pour chaque valeur clé, afin que seules les modifications les plus récentes soient répliquées au niveau suivant. Le programme Apply applique la modification la plus récente pour chaque ligne dans la table CCD au lieu d'appliquer une ligne pour chaque modification. Comme les tables condensées demandent des valeurs clés uniques pour chaque ligne, vous devez définir un index à entrées uniques.

**Remarque :** Avec une table source CCD condensée et complète, si la même ligne est mise à jour plusieurs fois dans un cycle Apply, une ligne supplémentaire, dupliquée avec la plus ancienne valeur de clé peut rester dans la table cible. Cette situation peut se produire car dans des sources CCD condensées, seule la dernière mise à jour parmi plusieurs mises à jour de clé demeure dans la table source CCD. Le programme Apply ne peut pas indiquer la valeur d'origine et, par conséquent, il lui est impossible de supprimer la ligne ayant la plus ancienne valeur de clé de la table cible.

4. Sachant que la table CCD est enregistrée, créez les tables de contrôle Capture dans la base de données de niveau intermédiaire si elles n'existent pas déjà.
5. Créez un ensemble d'abonnements entre le serveur de niveau 2 contenant la table CCD enregistrée et le serveur cible suivant (pour le niveau 3). Le programme Apply pour cet ensemble applique les modifications de la table CCD aux tables cible au niveau suivant. Le programme Apply utilise la table CCD pour effectuer une régénération intégrale et une réplication en mode capture des modifications. Normalement, vous utilisez un qualificatif Apply autre que celui ayant servi à remplir la table CCD, mais il est aussi possible d'employer le même.
6. Définissez le membre d'un ensemble d'abonnements mappant la table source CCD (niveau 2) et la table cible suivante (niveau 3). Vous pouvez configurer plusieurs membres avec des tables cible abonnées à cette table source CCD. S'il s'agit du niveau final dans votre configuration multi-niveaux, la table cible peut être de tout type. Toutefois, si vous envisagez d'avoir plus de trois niveaux, définissez la table cible de niveau 3 comme expliqué à l'étape 3 et répétez les étapes 4 à 5 pour ajouter d'autres niveaux.

**Remarque :** Dans le cas d'une régénération intégrale sur la table CCD externe (niveau intermédiaire), les programmes Apply pour tous les niveaux suivants utilisant cette table comme source réaliseront des régénérations intégrales. On parle alors de *régénération intégrale en cascade*.

### **Définition de cible en lecture-écriture (réplication bidirectionnelle)**

Dans le cadre d'une réplication bidirectionnelle, les modifications dans la table source maître sont répliquées sur des tables cible dépendantes, alors que les modifications dans les tables réplique peuvent être répliquées en retour sur la table source maître.

#### **Avant de commencer**

- Vous devez utiliser des contraintes d'intégrité référentielle déclaratives car aucun programme d'application ne met à jour à la fois les tables maîtres et réplique. Les violations d'intégrité référentielle ne sont pas détectables dans la logique applicative.
- Vous devez inclure dans les tables réplique toutes les contraintes référentielles figurant dans les tables maîtres afin d'empêcher des violations d'intégrité référentielle. Si vous omettez des contraintes référentielles, la mise à jour d'une table réplique peut entraîner une violation d'intégrité référentielle si elle est répliquée sur la table maître. Les outils d'administration ne copient pas de définitions de contraintes référentielles d'une table source dans des tables cible, pas plus qu'ils ne génèrent de nouvelles contraintes.
- Pour ignorer la vérification de l'intégrité référentielle lors d'une régénération intégrale, vous devez utiliser la routine d'exit ASNLOAD.

#### **Restrictions**

- Les types de table réplique cible ne sont pas pris en charge dans une configuration de journal de bord éloigné.
- Vous ne pouvez pas utiliser des tables CCD comme sources ou cibles dans une réplication bidirectionnelle.
- Pour permettre à des colonnes de type de données LOB de participer à une réplication bidirectionnelle, la valeur `CONFLICT_LEVEL` dans la table de registres doit être de 0.
- Les bases de données non DB2 ne peuvent pas posséder de types de table cible réplique et, par conséquent, ne peuvent pas prendre part à une réplication bidirectionnelle.

#### **A propos de cette tâche**

Dans une réplication bidirectionnelle, la table maître et ses répliques sont en lecture-écriture et agissent tant comme sources que comme cibles.

#### **Procédure**

Pour établir une configuration de réplication bidirectionnelle entre une table maître et une ou plusieurs tables réplique (chacune se trouvant dans une base de données distincte) :

1. Créez des tables de contrôle dans chaque base de données qui contiendra une table réplique si elles n'existent pas déjà.
2. Enregistrez la table source (la table maître) pour la réplication.
3. Créez un ensemble d'abonnements entre la base de données maître et la base de données cible qui contiendra une ou plusieurs répliques.

Si toutes les tables réplique se trouvent dans la même base de données et que toutes les tables maîtres figurent dans une autre, il vous faut un seul ensemble d'abonnements. Si les tables réplique se trouvent dans plusieurs bases de données en revanche, vous avez besoin d'autant d'ensembles d'abonnements que de bases de données réplique.

4. Définissez un membre d'un ensemble d'abonnements pour chaque mappage entre la table maître et sa table réplique associée.

Dans cette configuration, il n'existe qu'un programme Apply qui s'exécute généralement sur le serveur contenant les tables réplique. Le programme Apply pour cet ensemble extrait les changements depuis la table CD maître et les applique aux tables réplique. Il insère par ailleurs des modifications issues de la table CD de la table réplique et les applique à la table maître.

**Important :** Comme la table maître et les tables réplique dans des configurations de réplication bidirectionnelle répliquent des données entre elles, les tables réplique cible doivent contenir les mêmes colonnes que la table source. Vous pouvez créer une table réplique cible comportant un sous-ensemble de colonnes dans la table maître à condition que les colonnes manquantes soient définies comme admettant des valeurs NULL ou NOT NULL WITH DEFAULT sur le site maître, mais vous ne devez pas ajouter de nouvelles colonnes ou en renommer dans la réplique.

5. Définissez des propriétés source pour la table réplique. Lorsque vous créez un membre d'un ensemble d'abonnements avec une table réplique, la réplication SQL enregistre celle-ci comme source de réplication. Comme les tables réplique cible agissent comme des sources, elles possèdent des propriétés que vous pouvez définir en plus de celles habituelles de table cible, ce qui indique comment le programme Capture gère les modifications apportées à la réplique. Deux propriétés sont toutefois héritées de la table maître et ne sont pas modifiables pour la table réplique : le niveau de détection de conflit et si les régénérations intégrales sont désactivées. Le programme Capture pour cette source capture des modifications dans la table réplique et les stocke dans la table CD de la réplique.

**Important :** Même si les tables maîtres et réplique agissent comme des sources et des cibles, une copie avec régénération intégrale a uniquement lieu de la table maître dans la table réplique, et non l'inverse.

Pour éviter des conflits, vous devez prendre comme clé cible des tables réplique la même que la clé primaire de la table source maître ou l'index à entrées uniques. Comme la table maître peut mettre à jour les répliques et inversement, des conflits risquent de se produire si une mise à jour est effectuée dans une ligne de la table maître et une autre dans la même ligne d'une ou plusieurs tables réplique entre des cycles du programme Apply (les modifications figurent ainsi dans la table CD maître et la table CD réplique). Une table réplique hérite du niveau de détection de conflit d'une table ou vue source maître. Il est préférable de concevoir votre application de façon à ce qu'aucun conflit ne puisse se produire lorsque des données sont répliquées de la table maître sur toutes les tables réplique. Au moment d'enregistrer la table source maître, vous aviez le choix entre trois niveaux de détection de conflit.

Si vous avez défini des contraintes d'intégrité référentielle pour la table source, vous devez définir les mêmes pour la table réplique afin d'éviter des violations d'intégrité. Si une violation d'intégrité référentielle se produit, le cycle d'abonnement est automatiquement relancé.

## Utilisation d'une table existante en tant que table cible

Vous pouvez définir un membre d'ensemble d'abonnements pour insérer une table cible existante que vous avez définie hors de la réplication SQL.

Ce type de table cible définie par l'utilisateur peut être tout type de table valide pour la réplication (copie utilisateur, point de cohérence, agrégation de base ou des modifications, CCD ou réplique), sous réserve que la structure de la table soit valide. Par exemple, une table des points de cohérence définie par l'utilisateur doit inclure une colonne de type `TIMESTAMP` appelée `IBMSNAP_LOGMARKER`.

### Conditions requises

- Si la définition de membre d'ensemble d'abonnements contient moins de colonnes que dans la table cible existante, les colonnes de la table cible non utilisées par la réplication doivent accepter les valeurs Null ou être définies en tant que `NOT NULL WITH DEFAULT`.
- Il doit y avoir un index unique pour le point de cohérence, la copie utilisateur, la réplique et les tables CCD condensées. Lorsque vous définissez le membre d'ensemble d'abonnements à l'aide de la table cible existante, vous pouvez utiliser l'index unique ou en spécifier un nouveau.

### Restrictions

- Une définition de membre d'ensemble d'abonnements ne peut pas contenir davantage de colonnes qu'il n'en existe dans la table cible existante.
- Si vous utilisez le Centre de réplication, vous ne pouvez pas ajouter de colonne à un membre d'ensemble d'abonnements si cette colonne ne figure pas déjà dans la table cible.

La réplication vérifie les incohérences entre la table cible existante et la définition de membre d'ensemble d'abonnements.

**Important pour la configuration multiniveau :** si vous souhaitez créer une configuration multiniveau avec une table source en tant que niveau 1, une table CCD en tant que niveau 2 et une table existante en tant que niveau 3, définissez la table CCD pour qu'elle corresponde aux attributs spécifiés pour la table cible existante au moment de la définition du membre d'ensemble d'abonnements de la table cible existante entre le niveau 1 et le niveau 2. Ensuite, définissez un membre d'ensemble d'abonnements pour la table cible existante dans laquelle la table CCD est la table source.

## Propriétés communes à tous les types de tables cible

Vous pouvez définir des propriétés lorsque vous créez une table cible (quel que soit son type) en tenant compte de l'environnement de réplication souhaité.

Les rubriques suivantes expliquent les caractéristiques communes que vous pouvez définir pour le mappage entre les données sources et les tables cible.

### Réplication d'un sous-ensemble de colonnes source

Par défaut, la table cible contient toutes les colonnes source enregistrées, sauf les colonnes LOB. Il est possible de ne pas répliquer toutes ces colonnes et que la table cible ne prenne pas en charge tous les types de données définis dans la source.

Dans ce cas, sélectionnez uniquement les colonnes source à répliquer dans la table cible. Les colonnes enregistrées dans la table source et que vous ne sélectionnez

pas restent disponibles pour d'autres membres de l'ensemble d'abonnements, mais elles ne sont pas prises en compte pour le mappage source/cible.

Vous pouvez aussi ajouter des colonnes calculées à une table cible. Ces colonnes peuvent être définies par des fonctions scalaires SQL (comme SUBSTR) ou dérivées de colonnes, comme la division de la valeur de la colonne A par la valeur de la colonne B (colA/colB). Ces colonnes calculées peuvent faire référence à n'importe quelle colonne de la table source.

### Réplication d'un sous-ensemble de lignes source

Par défaut, la table cible contient toutes les lignes de la table source. Il est possible de ne pas répliquer toutes les lignes ou de répliquer seulement celles contenant plusieurs types de données vers diverses tables cible.

Vous pouvez définir un sous-ensemble (horizontal) de lignes dans le membre d'un ensemble d'abonnements qui contient des lignes correspondant à une certaine condition (une clause SQL WHERE).

Le prédicat SQL peut contenir des identificateurs standard ou délimités. Voir *DB2 SQL Reference* pour en savoir plus sur les clauses WHERE.

Par exemple, vous pouvez définir une clause WHERE afin de répliquer toutes les lignes pour un service d'une entreprise. Vous pouvez aussi définir une clause WHERE dans un membre d'un ensemble d'abonnements pour répliquer toutes les colonnes LOB (et la colonne de clé primaire) vers une table cible, ainsi qu'une clause WHERE dans un autre membre pour répliquer le reste des colonnes vers une table cible distincte. Dans ce cas, votre base de données cible peut comporter toutes les données de la table source ; dénormalisez toutefois la table source dans cette base de données pour adapter les performances de requête d'un entrepôt de données.

### Restrictions des prédicats de lignes

- N'entrez pas WHERE dans la clause (déjà implicite). Entrez uniquement WHERE dans la clause pour des instructions de sous-requête.
- Ne terminez pas la clause par un point-virgule (;).
- Si la clause WHERE contient l'expression booléenne OR, mettez le prédicat entre parenthèses ; par exemple, (COL1=X OR COL2=Y).
- Si la table cible est une table d'agrégation des modifications et qu'elle contient des colonnes image-avant, vous devez inclure celles-ci dans une clause GROUP BY.

### Exemples

Les exemples suivantes montrent les clauses WHERE utilisables pour filtrer des lignes de la table cible. Ces exemples sont très génériques et conçus pour servir de modèle.

#### Clause WHERE désignant des lignes avec des valeurs spécifiques

Pour copier uniquement les lignes contenant une valeur déterminée, comme MGR pour les responsables, utilisez une clause WHERE comme suit :

```
EMPLOYEE = 'MGR'
```

### Clause WHERE désignant des lignes avec un intervalle de valeurs

Pour copier uniquement les lignes dans un intervalle, comme les numéros d'employés compris entre 5000 et 7000 dans la table cible, utilisez une clause WHERE comme suit :

```
EMPID BETWEEN 5000 AND 7000
```

### Mappage entre colonnes source et colonnes cible

Par défaut, les noms de colonnes d'une table cible créée par la réplication SQL correspondent aux noms de colonnes de la table source. Vous pouvez modifier les noms et longueurs de la plupart des colonnes cible, ce qui ne vous empêche pas de les mapper à des colonnes source.

Vous pouvez modifier le nom de toutes les colonnes de vos tables cible, à l'exception des colonnes de contrôle de réplication (qui commencent par IBMSNAP ou par IBMQSQ). Si la table cible existe, le Centre de réplication mappe les colonnes par nom.

Les colonnes de table cible peuvent avoir des longueurs différentes de celles des colonnes source. Si la colonne cible porte un nom plus court que la colonne source, vous pouvez utiliser une expression dans le membre d'ensemble d'abonnements pour mapper les caractères du nom de la colonne la plus longue avec ceux du nom de la colonne la plus courte, ou encore enregistrer une vue incluant cette expression. Par exemple, si la colonne source est char(12) et que la colonne cible est char(4), vous pouvez utiliser l'expression suivante pour tronquer les valeurs de COL1 au cours de la réplication :

```
substr(col1, 1,4)
```

Si le nom de la colonne cible est le plus long, remplissez-le par des blancs.

**Remarque :** Certaines restrictions s'appliquent au mappage des colonnes LONG VARCHAR dans DB2 pour Linux, UNIX et Windows, dans DB2 pour z/OS et DB2 pour i5/OS.

### Utilisation du Centre de réplication

Lorsque vous créez une table cible à l'aide du Centre de réplication, vous pouvez renommer les colonnes au niveau de la cible, quel qu'en soit le type. Vous pouvez également modifier les attributs de colonne (type de données, longueur, échelle, précision, admission de valeurs Null), sous réserve que ces attributs soient compatibles.

Vous ne pouvez pas utiliser le Centre de réplication pour renommer les colonnes de tables cible existantes. Si les colonnes source et cible ne correspondent pas, vous pouvez utiliser le Centre de réplication pour mapper les colonnes de la table source et de la table cible, ou encore créer une vue de la table cible contenant une correspondance avec les noms de colonnes source.

### Mappage avec des tables relationnelles non DB2

Si vous mappez une table DB2 à une table relationnelle non DB2 portant un pseudonyme, les types de données de certaines colonnes risquent de ne pas être compatibles. Si les types de données des colonnes source ne sont pas compatibles avec ceux des colonnes cible, vous pouvez modifier le type de données de la cible pour le rendre compatible avec la source :

- Vous pouvez ajouter des colonnes calculées pour ajuster les types de données de la source, afin qu'ils correspondent au type de données requis pour la cible.
- Vous pouvez modifier le pseudonyme d'une table cible relationnelle non DB2, pour modifier les conversions de type de données.

**Exemple** : vous souhaitez répliquer les données d'une table source DB2 contenant une colonne DB2 dont le type de données est DATE dans une table cible Oracle contenant une colonne dont le type de données est DATE.

Tableau 5. Mappage d'une colonne DB2 portant le type de données DATE à une colonne Oracle portant le même type

Colonne DB2	Mappage de données de pseudonymes	Colonne Oracle
A_DATE DATE	A_DATE TIMESTAMP A_DATE DATE	A_DATE DATE

La table cible Oracle est créée avec le type de données DATE (qui peut contenir des données de date et d'horodatage). Le pseudonyme initial du type de données Oracle DATE d'une base de données fédérée correspond au type de données DB2 TIMESTAMP. Le Centre de réplication DB2 et les commandes système System i utilisées pour la réplication transforment le type de données de pseudonyme en DATE ; par conséquent, c'est le type DATE qui est répliqué dans Oracle, et non le type TIMESTAMP.

### Clé cible

Lorsqu'une table cible condensée est impliquée dans une réplication en mode capture des modifications, le programme Apply l'oblige à posséder une clé primaire ou un index à entrées uniques, à savoir une *clé cible*.

Vous pouvez choisir les colonnes à utiliser comme index à entrées uniques pour votre table cible. Les types suivants de tables cible sont condensés et requièrent une clé cible :

- Copie utilisateur
- Point de cohérence
- Réplique
- CCD condensée

Si vous créez une table cible, 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.

Le nom par défaut est celui du profil de l'objet cible pour le serveur cible, le cas échéant. Si ce profil n'est pas défini, le nom par défaut est IX, suivi du nom de la table cible. Par exemple, si la table cible se nomme TGEMPLOYEE, le nom de son index est par défaut IXTGEMPLOYEE.

### Options des index à entrées uniques

Selon si vous créez une table cible ou en utilisez une existante, les options de création d'index à entrées uniques changent.

#### Nouvelle table cible

Pour créer un index à entrées uniques pour une nouvelle table cible, vous disposez de deux options :

- Indiquez les colonnes souhaitées comme index à entrées uniques pour la table cible.

- Laissez la réplication SQL choisir un index à entrées uniques.  
Si vous ne sélectionnez pas de colonnes pour l'index à entrées uniques, la réplication SQL recherche dans la table source l'une des définitions ci-après, dans l'ordre suivant :
  1. Une clé primaire
  2. Une contrainte d'unicité
  3. Un index à entrées uniques

Si la réplication SQL trouve l'une de ces définitions pour la table source et que les colonnes associées sont enregistrées et font partie de la table cible, la réplication SQL se sert de la clé primaire de la table source (ou index à entrées uniques ou du numéro relatif d'enregistrement) comme clé cible. En cas de contrainte d'unicité, la réplication SQL crée un index à entrées uniques pour la table cible en utilisant les colonnes de contrainte.

**System i** Pour une table source System i ne comportant pas de clé primaire ou d'index à entrées uniques, modifiez l'enregistrement afin d'utiliser le numéro relatif d'enregistrement (RNN) comme facteur d'unicité. Lorsque vous définissez le numéro d'un membre d'un ensemble d'abonnements, indiquez la colonne de numéro relatif d'enregistrement comme index à entrées uniques pour la table cible.

**System i** Pour les tables cible sur des systèmes System i employant le numéro relatif d'enregistrement comme clé cible, vous devez exécuter le programme Apply sur System i afin de répliquer sur ces tables cible.

#### Table cible existante

Pour les tables cible existantes, vous devez sélectionner l'index à entrées uniques. Vous disposez des options suivantes :

- Utilisez un index qui existe déjà pour la table cible.  
Pour ce faire, sélectionnez les colonnes désignant l'index dans le Centre de réplication. Si le Centre de réplication trouve une correspondance exacte, il définit uniquement une clé cible pour le programme Apply ; sinon, il crée l'index à entrées uniques et définit une clé cible pour le programme Apply.
- Créez un autre index pour la table cible.  
L'index à entrées uniques sera créé s'il n'existe pas déjà et la clé cible sera définie pour le programme Apply.

**Important :** Si vous sélectionnez une clé pour la table cible comportant des colonnes actualisables dans la table source, vous devez commander au programme Apply d'effectuer des mises à jour spéciales dans les colonnes de clé cible.

#### Mode de mise à jour par le programme Apply des colonnes de clé cible avec l'option de changement de clé cible

Si vous choisissez l'option de changement de clé cible lorsque vous définissez un membre d'un ensemble d'abonnements, le programme Apply effectue des mises à jour spéciales des colonnes de clé cible au moment où la clé cible change.

#### Condition préalable

Pour que le programme Apply mette à jour des colonnes de clé cible, les colonnes source intégrées à cette clé doivent être enregistrées avec les colonnes image-avant dans la table CD (ou CCD). Si vous n'avez pas défini l'enregistrement source pour



capturer les valeurs image-avant des colonnes composant la clé cible, vous devez modifier cet enregistrement afin de les inclure avant l'abonnement à une table cible avec une autre clé.

### Restrictions

- Vous ne pouvez pas utiliser l'option de changement de clé cible pour les tables source enregistrées en vue de capturer des mises à jour comme paires de suppression/insertion.
- Vous ne pouvez pas mapper une expression dans une table source vers une colonne clé dans une table cible si le programme Apply met à jour cette table en fonction des images-avant de la colonne de clé cible (si la colonne TARGET\_KEY\_CHG de la table IBMSNAP\_SUBS\_MEMBR comporte la valeur Y pour cette table cible).

Si les valeurs image-avant des colonnes de clé cible se trouvent dans la table CD (ou CCD), sélectionnez l'option de membre d'un ensemble d'abonnements pour que le programme Apply emploie ces valeurs au moment de la mise à jour des colonnes de clé cible.

Si vous ne choisissez pas le programme Apply pour l'utilisation des valeurs d'images-avant lors de la mise à jour de colonnes clés cibles, la réplication SQL ne répliquera pas correctement les données lors de la mise à jour des colonnes de la table source appartenant à la clé cible.

Le programme Apply tente de mettre à jour avec la nouvelle valeur la ligne dans la table cible, mais il ne trouve pas la nouvelle valeur clé. Il convertit alors la mise à jour en une insertion et insère la nouvelle valeur clé dans la table cible. Dans ce cas, l'ancienne ligne avec la valeur clé antérieure reste dans la table cible et devient inutile.

Lorsque vous précisez que les modifications des colonnes de clé cible doivent être traitées avec des valeurs image-avant, le programme Apply peut identifier la ligne comportant la valeur clé antérieure et la mettre à jour avec les nouvelles valeurs. Par exemple, si la variable *chgt\_clé\_cible* a la valeur N, l'instruction SQL pour l'opération de mise à jour est :

```
UPDATE
targettable SET <colonnes non clé>= valeurs image-après
WHERE
<colonnes de clé> = valeur image-après
```

Si la variable *chgt\_clé\_cible* a la valeur Y en revanche, l'instruction SQL pour l'opération de mise à jour est :

```
UPDATE
targettable SET <toutes les colonnes> = valeurs
image-après
WHERE <colonnes de clé> = valeurs
image-après
```



---

## Chapitre 6. Réplication de types spéciaux de données dans la réplication SQL

Lorsque vous répliquez des types spéciaux de données (comme LOB, ROWID ou non DB2), vous devez prendre en compte certaines conditions et restrictions. Dans certains cas, vous devrez éventuellement suivre d'autres étapes de configuration pour que la réplication SQL fonctionne avec ces types de données.

Les rubriques suivantes offrent des informations sur la réplication de types spéciaux de données :

---

### Restrictions générales de données pour la réplication SQL

La réplication SQL présente des restrictions spécifiques pour certains types de données, y compris les restrictions sur le chiffrement de données et les restrictions sur le type de données.

#### Restrictions sur le chiffrement des données

La réplication SQL peut répliquer certains types de données chiffrées.

##### EDITPROC

La réplication SQL prend en charge les tables source DB2 pour z/OS définies avec une routine de modification (EDITPROC) afin d'augmenter la sécurité des données. Pour que ces tables puissent être utilisées comme sources de réplication, 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.

#### Chiffrer une fonction scalaire dans DB2 for Linux, UNIX et Windows

Les données de colonne peuvent être chiffrées et déchiffrées en utilisant la fonction de chiffrement scalaire dans DB2 for Linux, UNIX et Windows. Pour pouvoir l'utiliser avec la réplication, 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 SQL prend en charge les colonnes définies dans des tables DB2 for 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

La réplication SQL ne peut pas répliquer les types de données suivants :

- XML DB2
- Colonnes d'objets LOB de sources relationnelles non DB2
- 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 réplication 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.

La réplication SQL peut répliquer les types de données suivants dans certaines circonstances :

- Données variables longues (LONG VARGRAPHIC) si les tables source et cible résident dans DB2 pour z/OS.
- Données à caractère de variable longue (LONG VARCHAR et LONG VARGRAPHIC) exigeant que les tables de base de données source se trouvent dans DB2 for z/OS ou que les tables source et cible soient dans DB2 for Linux, UNIX et Windows. 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 réplication. Si vous ajoutez des colonnes LONG VARCHAR à la table une fois qu'elle est enregistrée en tant que source et que cette 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.

La réplication SQL ne peut pas répliquer une table contenant des types de données abstraits.

La réplication SQL peut répliquer des tables contenant des colonnes de types de données spatiales, mais ne peut pas répliquer les colonnes elles-mêmes.

Les types de données définis par l'utilisateur (types de données distincts dans DB2) sont convertis en type de données de base dans la table de modification des données (CD) avant d'être répliqués. En outre, si la table cible est créée par la réplication SQL comme faisant partie intégrante de la définition de membre d'ensemble d'abonnements, les types de données définis par l'utilisateur sont convertis en type de données de base dans la table cible et dans la table de modification des données (CD).

---

## Types de données d'objets LOB

La réplication SQL supporte des types de données d'objets LOB, dont des objets BLOB, des objets CLOB et des objets DBCLOB.

Cette rubrique fait référence aux types de données BLOB, CLOB et DBCLOB en tant que données LOB.

Le programme Capture lit le descripteur d'objet LOB dans les enregistrements de journal si des données dans la colonne LOB ont changé et demandent donc une réplication. Il ne copie en revanche pas les données LOB dans les tables de modification des données (CD). Si une colonne LOB change, le programme Capture définit un indicateur dans la table CD. Lorsque le programme Apply lit cet indicateur, il copie l'intégralité de la colonne LOB (et pas seulement les parties modifiées) directement de la table source à la table cible.

Sachant qu'une colonne LOB peut contenir jusqu'à deux gigaoctets de données, vous devez vérifier que la bande passante réseau est suffisante pour le programme Apply. Par ailleurs, les tables cible doivent avoir assez d'espace disque pour accueillir les données LOB.

#### **Restrictions :**

- Le programme Apply copie toujours la version la plus actuelle d'une colonne LOB directement depuis la table source (et non depuis la table CD), même si cette colonne est plus récente que d'autres dans la table CD. Par conséquent, si la colonne LOB change dans la ligne cible, il se peut qu'elle ne soit pas cohérente avec le reste des données dans cette ligne. Pour réduire le risque d'incohérence des données dans la ligne cible, vérifiez que l'intervalle entre les cycles du programme Apply est aussi court qu'approprié pour votre application.
- Vous pouvez répliquer jusqu'à 15 colonnes LOB par table. Si vous enregistrez une table avec plus de 15 colonnes LOB, le programme Apply renvoie un message d'erreur. Le Centre de réplication renvoie un message d'erreur si vous tentez d'enregistrer plus de 15 colonnes LOB par table.
- Vous pouvez copier des données LOB dans les tables réplique, à condition que la détection de conflit soit désactivée.
- Pour copier des données LOB entre DB2 pour OS/390 Version 6 (ou ultérieure) et DB2 pour Linux, UNIX, et Windows, vous avez besoin de DB2 Connect Version 7 ou ultérieure.
- Vous ne pouvez pas faire référence à des données LOB à l'aide de pseudonymes.
- Les valeurs image-avant pour les colonnes LOB ou ROWID ne sont pas prises en charge.
- La réplication n'est pas prise en charge pour DB2 Extensions pour les extensions texte, audio, vidéo, image ou autres si des fichiers de contrôle supplémentaires associés aux données de la colonne LOB de l'extension sont conservés hors de la base de données.
- LA réplication SQL ne peut répliquer qu'un objet LOB complet. Elle ne peut pas répliquer des parties d'un objet LOB.
- Vous ne pouvez pas répliquer des colonnes LOB si vous utilisez la configuration de journal distante dans votre environnement de réplication sur System i.
- Lorsque vous utilisez des objets LOB dans une réplication bidirectionnelle, vous devez définir le niveau de conflit sur 0.

---

## **Réplication de nouveaux types de données DB2 version 9.7 (Linux, UNIX, Windows)**

La réplication SQL prend en charge de nouveaux types de données introduits avec la version 9.7 de DB2 pour Linux, UNIX et 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»
- «NUMBER», à la page 107

## TIMESTAMP et précision étendue

La réplication SQL prend en charge la réplication de données TIMESTAMP avec une précision étendue, allant de TIMESTAMP(0) à TIMESTAMP(12). Vous pouvez mapper des colonnes dont la précision ne correspond pas. Si les bases de données source et cible et les programmes Capture et Apply sont de version 9.7 ou une version ultérieure, la données source est complétée ou tronquée sur la base de données cible.

Dans un environnement de niveau mixe où seule la base de données DB2 source est de version 9.7, les colonnes TIMESTAMP peuvent également devoir être complétées ou tronquées. La réplication de ces colonnes ne peut se faire que lorsque la version des programmes Capture et Apply est la version 9.7 ou une version ultérieure. Par exemple, si vous avez répliqué une source V9.7 vers une cible V9.5 et qu'une table enregistrée comprend une colonne TIMESTAMP(12), le programme Apply V9.7 tronque six chiffres de la portion concernant les fractions de secondes de la valeur TIMESTAMP. La troncature est nécessaire car DB2 version 9.5 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 la précision V9.7 par défaut de TIMESTAMP(6). tableau 6 affiche la valeur pour la source et la valeur tronquée obtenue pour la cible.

**z/OS** **Remarque :** Lors de la gestion de ces nouveaux types de données, SQL 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.

Tableau 6. 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

Si la base de documents cible est antérieure à la version 9.7, les valeurs TIMESTAMP avec une précision inférieure à la précision TIMESTAMP(6) par défaut sont automatiquement complétées par DB2 pour que la portion relative aux fractions de secondes contiennent six caractères.

## 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 SQL 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 SQL, 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 SQL prend en charge DECFLOAT et DECIMAL, 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 SQL 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, effectuez les opérations d'enregistrement et d'abonnement comme d'habitude. Les tables CD et cible sont créées 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 des tables CD et cible ne peuvent 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 configuration de réplication lors de la définition du membre d'un ensemble d'abonnements. 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 SQL. 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 7. Sous-ensemble de données dans un environnement de réplication SQL

La réplication implique en général la définition de sous-ensembles. Elle peut aussi demander de choisir certaines colonnes et lignes d'une table source au moment d'enregistrer une source de réplication. Vous devrez éventuellement choisir également des colonnes enregistrées à répliquer sur chaque table cible lorsque vous créez des ensembles d'abonnements.

En fonction de vos besoins de réplication, vous pouvez définir un sous-ensemble de données dans la source lors de l'enregistrement ou dans la cible pendant l'abonnement :

- Si vous ne possédez qu'une cible pour une source ou si plusieurs cibles requièrent les mêmes données, il est possible de définir un sous-ensemble ou de manipuler des données lors de l'enregistrement, puisqu'il est inutile de prendre en compte les besoins éventuellement différents de cibles diverses.
- Si vous avez une source et plusieurs cibles et que ces dernières ont des besoins différents pour les données à appliquer, vous ne pourrez peut-être pas définir un sous-ensemble au moment de l'enregistrement. Dans ce cas, vous définissez le sous-ensemble de données lors de l'abonnement.

Les vues permettent de définir des sous-ensembles de données lors de l'enregistrement et les prédicats de requête permettent de définir des sous-ensembles de données lors de l'abonnement. Dans de nombreux cas, le choix de l'utilisation de prédicats d'abonnement ou de vues enregistrées dépend de vos préférences. Certains facteurs peuvent vous influencer :

- Des vues peuvent déjà exister et répondre aux qualifications requises pour les vues enregistrées pour la réplication.
- Les vues peuvent être une approche plus simple pour vérifier les sous-ensembles définis pour la réplication.
- Les prédicats d'abonnement sont stockés dans des tables de contrôle de réplication et lorsqu'ils sont utilisés, il n'est plus nécessaire de créer et de gérer des vues.

N'employez aucune de ces techniques si vous répliquez sur des tables réplique cible. La table maître et les tables réplique dans des configurations de réplication bidirectionnelle répliquent des données entre elles. Les tables réplique peuvent inclure un sous-ensemble des colonnes de la table source si les colonnes inutilisées admettent des valeurs NULL. Sinon, les tables réplique doivent comporter les mêmes colonnes que la table source, ce qui vous empêche de définir des sous-ensembles de colonnes et d'ajouter ou de renommer des colonnes.

---

### Etablissement de sous-ensembles de données pendant l'enregistrement

Certaines techniques avancées sont utiles lors de l'établissement de sous-ensembles de données avant ou après leur capture dans une source enregistrée. Ces techniques sont particulièrement utiles lorsque vous souhaitez capturer un sous-ensemble de données et le répliquer dans de nombreuses tables cible.

Vous pouvez établir des sous-ensembles de données avant ou après leur capture dans une source enregistrée. Les techniques décrites dans cette section peuvent être utilisées dans toutes les configurations de réplication, à l'exception de la réplication bidirectionnelle ou de la réplication entre homologues.

L'établissement de sous-ensembles pendant l'enregistrement peut augmenter les performances de réplication, car cela permet de diminuer la quantité de données ajoutée par le programme Capture à la table de modification des données, ainsi que la quantité lue par le programme Apply. Cela permet également de réduire l'espace de stockage, car un nombre plus réduit de lignes figure dans la table de modification des données.

## **Création de sous-ensembles de données source à l'aide de vues**

Lorsque vous enregistrez une source, vous sélectionnez les colonnes que vous souhaitez utiliser pour la réplication. Les colonnes sélectionnées sont capturées en vue de la réplication. Dans certains cas, une fois que vous avez enregistré une source en vue de la réplication, vous pouvez enregistrer une vue de cette source.

Par exemple, supposons que le service des Ressources humaines gère une table contenant les données relatives au personnel, y compris les informations sur les salaires. Pour pouvoir gérer une base de données de sauvegarde, la table entière est enregistrée et abonnée à un site de sauvegarde. Toutefois, si un autre site cible souhaite s'abonner à cette table, vous pouvez masquer les informations sur les salaires, qui ne s'afficheront pas pour le deuxième abonné. La solution consiste à enregistrer une vue de la table, puis à octroyer au deuxième abonné des droits d'accès sur la vue enregistrée uniquement ; de cette façon, il ne peut pas accéder aux informations sur les salaires. Vous pouvez créer un abonnement sur cette vue enregistrée.

Vous pouvez également enregistrer des vues qui contiennent deux tables source ou davantage. Par exemple, si vous possédez une table de clients et une table de succursales et que vous souhaitez établir des sous-ensembles de clients dans la cible, l'unique méthode efficace consiste à joindre les deux tables, afin que seuls les clients d'une succursale spécifique soient répliqués dans une cible spécifique. Dans ce cas, vous devez éviter d'utiliser les suppressions doubles.

## **Définition des déclencheurs dans des tables CD pour empêcher la capture de lignes spécifiques**

Dans certains scénarios de réplication, vous pouvez empêcher la capture de modifications dans des lignes et leur réplication sur des tables cible. Pour empêcher la capture de modifications, définissez des déclencheurs dans vos tables CD.

Lorsque vous enregistrez une source, les outils d'administration vous permettent de sélectionner les colonnes à capturer, mais vous ne pouvez pas empêcher la réplication de certaines modifications dans ces lignes. Dans certains scénarios de réplication, vous pouvez empêcher la capture de modifications dans des lignes et leur réplication sur des tables cible. Par exemple, si vous voulez que les tables cible contiennent toutes les lignes et qu'aucune d'elles ne doit être supprimée, vous ne souhaitez pas que des suppressions soient répliquées depuis la source.

Pour supprimer la capture de certaines modifications, définissez des déclencheurs sur vos tables CD. Ces déclencheurs indiquent les modifications que le programme

Capture doit ignorer, en empêchant l'ajout de lignes correspondant aux modifications apportées dans la table CD. Vous ne pouvez pas créer ces déclencheurs avec le Centre de réplication, mais vous pouvez les générer manuellement pour une table CD existante (une fois la source enregistrée). Le programme Capture ignore tout échec du déclencheur montrant SQLSTATE avec la valeur 99999 et la ligne n'est pas insérée dans la table CD.

Imaginez par exemple vouloir que toutes les opérations DELETE de la table source soient supprimées pendant la réplication depuis la table SAMPLE.TABLE (la table CD°. Le déclencheur suivant empêche les lignes correspondant à des opérations DELETE d'être insérées dans la table CD :

```
CREATE TRIGGER SAMPLE.CD_TABLE_TRIGGER
NO CASCADE BEFORE INSERT ON SAMPLE.CD_TABLE
REFERENCING NEW AS CD
FOR EACH ROW MODE DB2SQL
WHEN (CD.IBMSNAP_OPERATION = 'D')
SIGNAL SQLSTATE '99999' ('CD INSERT FILTER')
```

Vous pouvez ajouter l'instruction create trigger à l'instruction SQL générée lors de l'enregistrement. Vous devez alors exécuter l'instruction SQL modifiée pour terminer l'enregistrement et créer les déclencheurs dans les tables CD.

Ces déclencheurs s'exécutent chaque fois que le programme Capture tente d'insérer une ligne dans la table CD. Vous devez déterminer si l'utilisation de déclencheurs vous offrira les meilleures performances dans votre configuration de réplication. Vous pouvez augmenter ou diminuer le rendement de données en ajoutant des déclencheurs aux tables CD. Utilisez des déclencheurs dans la table CD pour supprimer un nombre important de modifications dans la source. Si vous envisagez de capturer la plupart des modifications mais souhaitez empêcher la réplication de certaines, vous pouvez supprimer les lignes superflues lors de l'abonnement.

---

## Définition de sous-ensembles de données pendant l'abonnement

La définition de sous-ensembles de données pendant l'abonnement peut améliorer les performances de réplication en réduisant la quantité de données extraites par le programme Apply. Moins les tables cible comportent de lignes et plus les besoins en mémoire sont limités.

Le programme Apply utilise des prédicats pour identifier les données à copier lors d'une régénération intégrale et d'une réplication en mode capture des modifications. Le Centre de réplication et ASNCLP vous permettent d'indiquer des valeurs de prédicats pour une régénération intégrale et une réplication en mode capture des modifications. Vous pouvez ajouter d'autres informations de prédicats concernant seulement la réplication en mode capture des modifications car elles ne sont pas disponibles pendant une régénération intégrale. Vous devez ajouter ces informations dans la colonne UOW\_CD\_PREDICATES de la table IBMSNAP\_SUBS\_MEMBR via SQL.

Imaginez par exemple une table enregistrée ALL.CUSTOMERS et la table CD associée ALL.CD\_CUSTOMERS. Imaginez aussi vouloir que la cible d'abonnement ne contienne qu'un sous-ensemble de ALL.CUSTOMERS, avec la colonne ACCT\_BALANCE supérieure à 50000, et que vous souhaitez conserver les données d'historique dans la table cible (aucune donnée ne doit être supprimée de la table cible). Vous pouvez alors créer le membre d'un ensemble d'abonnements avec une valeur PREDICATES de 'ACCT\_BALANCE > 50000'.

Vous ne pouvez pas utiliser le Centre de réplication ou ASNCLP pour empêcher les suppressions de la table cible car les informations sur le type d'opération sont stockées dans la table CD et ne sont pas disponibles dans la table ou la vue source. Par conséquent, vous devez générer un autre prédicat de réplication en mode capture des modifications à l'aide d'une instruction SQL incluant les informations suivantes. En fonction de votre scénario, vous devez éventuellement ajouter des colonnes à l'instruction update afin de mettre à jour une ligne dans la table IBMSNAP\_SUBS\_MEMBR :

```
UPDATE ASN.IBMSNAP_SUBS_MEMBR SET UOW_CD_PREDICATES = 'IBMSNAP_OPERATION <>'D''
WHERE APPLY_QUAL = 'qual_apply' AND
SET_NAME = 'nom_ensemble' AND
SOURCE_OWNER = 'ALL' AND SOURCE_TABLE = 'CUSTOMERS'
```

Vous devez configurer manuellement la colonne UOW\_CD\_PREDICATES pour le prédicat d'un membre d'un ensemble d'abonnements faisant référence à une colonne non disponible lors d'une régénération intégrale (y compris les colonnes image-avant dans la table CD), les colonnes de temps système de la table CD et toute colonne de la table UOW).

Par défaut, le programme Apply ne joint pas les tables UOW et CD pour des tables cible de copie utilisateur : il extrait et applique en effet directement les données depuis la table CD. Si le prédicat doit faire référence à la table UOW et que la table cible est une table de copie utilisateur, vous devez définir la valeur de la colonne JOIN\_UOW\_CD à Y dans la table IBMSNAP\_SUBS\_MEMBR. La définition de cet indicateur assure que le programme Apply joint les tables UOW et CD.

Pour indiquer des prédicats dépassant 1024 octets (capacité de la colonne PREDICATES de la table IBMSNAP\_SUBS\_MEMBR) pour un sous-ensemble de lignes, vous devez utiliser une vue source.

Si vous utilisez des instructions de prédicats complexes pour un ensemble d'abonnements, mettez toute l'expression entre parenthèses. Par exemple, avec les clauses AND et OR dans une instruction de prédicat, entrez l'expression comme suit :

```
((TOSOURCE = 101 AND STATUS IN (202,108,109,180,21,29,32,42))
OR (SOURCE = 101))
```

---

## Chapitre 8. Manipulation de données dans un environnement de réplication SQL

Vous pouvez transformer ou étendre vos sources de données avant de les répliquer vers des tables cible.

Par exemple, vous pouvez manipuler des données de l'une des façons suivantes :

- Réaliser un nettoyage des données
- Effectuer un regroupement des données
- Remplir des colonnes de la table cible n'existant pas dans la table source

Servez-vous du programme Apply pour manipuler des données, tant avant qu'après leur application à la cible, de l'une des façons suivantes :

- Utilisation de procédures stockées ou d'instructions SQL
- «Mappage des colonnes source et cible portant des noms différents», à la page 115
- «Création de colonnes calculées», à la page 115

Vous pouvez manipuler des données avant ou après leur capture. Manipulez vos données à l'enregistrement plutôt qu'à l'abonnement si vous voulez les utiliser une fois et répliquer les données transformées vers de nombreuses tables cible. Manipulez vos données pendant l'abonnement plutôt que l'enregistrement pour capturer toutes les données source et appliquer de façon sélective les données transformées à des cibles individuelles.

Dans certains scénarios de réplication, vous pouvez manipuler le contenu des données source stockées dans la table CD. Un déclencheur, une expression lors de l'abonnement ou une vue source peuvent servir à réaliser la même tâche. Chaque méthode présente des avantages et des inconvénients. Un déclencheur peut s'avérer trop coûteux en termes de cycles d'UC utilisés. Une vue vous permet de configurer les fonctions en une fois au lieu d'intervenir lors de plusieurs abonnements.

Par exemple, si une valeur déterminée manque dans la table source, vous ne souhaitez pas que le programme Capture capture des valeurs NULL.

Vous pouvez alors utiliser des déclencheurs sur votre table CD pour préciser les conditions dans lesquelles le programme Capture étend les données au moment de les intégrer à la table CD. Dans ce cas, vous pouvez indiquer que le programme Capture doit insérer une valeur par défaut dans la table CD lorsqu'il rencontre une valeur NULL dans la source. Vous pouvez utiliser le code suivant pour créer un déclencheur fournissant une valeur par défaut sans ambiguïté si des données manquent après la mise à jour de la table source :

```
CREATE TRIGGER ENHANCECD
NO CASCADE BEFORE INSERT ON CD_TABLE
REFERENCING NEW AS CD
FOR EACH ROW MODE DB2SQL
WHEN (CD.COL1 IS NULL)
SET CD.COL1 = 'MISSING DATA'
END
```

Au lieu d'un déclencheur, vous pouvez recourir à la fonction scalaire COALESCE de DB2 dans une vue source enregistrée ou dans une expression de l'abonnement. Dans une vue enregistrée, la fonction de coalescence renvoie la première valeur définie.

#### Exemple partiel utilisant une vue source

```
CREATE VIEW
SAMPLE.SRCVIEW (colonnes) AS SELECT
... COALESCE(A.COL1, 'MISSING DATA') ...
FROM SAMPLE.TABLE A
```

#### Exemple partiel utilisant une expression

```
COALESCE(CD.COL1, 'MISSING DATA')
```

---

## Extension des données à l'aide de procédures stockées ou d'instructions SQL

Lorsque vous définissez les informations d'un ensemble d'abonnements, vous pouvez aussi déterminer des instructions de processus d'exécution via des instructions SQL ou des procédures stockées que le programme Apply doit exécuter chaque fois qu'il traite un ensemble spécifique. Ces processus d'exécution permettent de manipuler les données lors de la réplication.

Ces instructions sont utiles pour supprimer des tables CCD et contrôler la séquence de traitement des ensembles d'abonnements. Vous pouvez exécuter les instructions de traitement d'exécution sur le serveur de contrôle de Capture avant le traitement d'un ensemble d'abonnements ou sur le serveur cible avant ou après le traitement d'un ensemble d'abonnements. Par exemple, vous pouvez exécuter des instructions SQL avant d'extraire les données et/ou après les avoir répliquées sur des tables cible.

**Restriction pour les pseudonymes :** Les tables DB2 fédérées (qui utilisent des pseudonymes) sont généralement mises à jour dans une même unité de travail. Lorsque vous ajoutez une instruction SQL à un ensemble d'abonnements s'exécutant après que le programme Apply a appliqué toutes les données aux cibles, vous devez faire précéder cette instruction d'une instruction SQL COMMIT dans l'une des situations suivantes :

- L'instruction SQL insère, met à jour ou supprime un pseudonyme sur un serveur autre que celui où figurent les tables cible ou les pseudonymes cible pour l'ensemble d'abonnements.
- L'instruction SQL insère, met à jour ou supprime une table locale du serveur de contrôle Apply, mais les pseudonymes cible pour l'ensemble d'abonnements se trouvent sur un serveur distant.

L'instruction COMMIT valide le travail du programme Apply avant de traiter l'instruction SQL ajoutée.

Les procédures stockées emploient l'instruction SQL CALL sans paramètres. Le nom de la procédure doit compter un maximum de 18 caractères (pour System i, ce chiffre est de 128). Si la table source ou cible est dans une base de données relationnelle non DB2, les instructions SQL sont exécutées par rapport à la base de données DB2 fédérée. Les instructions SQL ne sont jamais exécutées par rapport à une base de données non DB2. Les procédures d'exécution de chaque type sont exécutées en même temps dans une même transaction. Vous pouvez aussi définir des valeurs SQLSTATE acceptable pour chaque instruction.

Utilisez la routine d'exit ASNDONE pour manipuler des données au terme du traitement de chaque ensemble (et non après le traitement d'un ensemble spécifique).

---

## Mappage des colonnes source et cible portant des noms différents

Lorsque vous utilisez le Centre de réplication ou le programme de ligne de commande ASNCLP pour définir un membre d'un ensemble d'abonnements et que la table cible référencée n'existe pas, vous pouvez renommer des colonnes de la table cible, quel que soit le type de celle-ci. Vous pouvez également modifier des attributs de colonne compatibles.

Vous pouvez aussi modifier des attributs de colonne (type de données, longueur, échelle, précision et acceptabilité des valeurs indéfinies NULL) lorsqu'ils sont compatibles. Vous ne pouvez pas utiliser les outils d'administration de réplication pour renommer des colonnes de tables cible existantes.

Les outils d'administration tentent de mapper des colonnes par nom si la table cible référencée par le membre d'un ensemble d'abonnements existe. Si les colonnes source et cible ne correspondent pas, vous pouvez soit employer les outils pour les mapper de la source à la cible, soit créer une vue de la table cible contenant une correspondance des noms de colonnes source.

---

## Création de colonnes calculées

Bien que vous ne puissiez pas modifier les noms des colonnes des tables cible existantes, vous pouvez modifier les expressions des colonnes source, afin qu'elles correspondent parfaitement aux colonnes des tables cible existantes (ou qu'elles soient compatibles).

### Avant de commencer

Lorsque vous créez des expressions faisant référence à des colonnes de tables source, préfixez le nom de la colonne source de deux points (:), et ajoutez un espace après le nom de la colonne. Par exemple, :COL1 .

Vous pouvez obtenir de nouvelles colonnes à partir des colonnes source existantes, à l'aide des expressions SQL. Pour les types de table cible d'agrégats, vous pouvez définir de nouvelles colonnes via l'utilisation des fonctions d'agrégat telles que COUNT ou SUM. Pour les autres types de tables cible, vous pouvez définir de nouvelles colonnes en utilisant des fonctions scalaires dans des expressions. Si les colonnes des tables source et cible ne diffèrent que par leur nom mais qu'elles sont par ailleurs compatibles, vous pouvez utiliser le Centre de réplication ou le programme ASNCLP pour mapper une colonne à une autre.

Par exemple, supposons que vous ayez une table source (SRC.TABLE) et une table cible (TGT.TABLE) :

```
CREATE TABLE SRC.TABLE (SRC_COL1 CHAR(12) NOT NULL, SRC_COL2 INTEGER,  
SRC_COL3 DATE, SRC_COL4 TIME, SRC_COL5 VARCHAR(25))  
CREATE TABLE TGT.TABLE (TGT_COL1 CHAR(12) NOT NULL,  
TGT_COL2 INTEGER NOT NULL, TGT_COL3 TIMESTAMP, TGT_COL4 CHAR(5))
```

Utilisez la procédure suivante pour mapper la table cible souhaitée à l'aide de colonnes calculées au cours de l'enregistrement :

1. Utilisez le Centre de réplication pour mapper la colonne SRC\_COL1 de la table source à la colonne TGT\_COL1 de la table cible. Etant donné que ces colonnes sont compatibles, il est inutile d'utiliser une expression pour les mapper.
2. Utilisez l'expression COALESCE(:SRC\_COL2, 0) pour calculer les valeurs de colonne et pour effectuer le mappage permettant d'obtenir TGT\_COL2. SRC\_COL2 admet les valeurs Null, alors que TGT\_COL2 ne les admet pas ; par conséquent, vous devez exécuter cette procédure pour vous assurer qu'une valeur NOT NULL est indiquée pour la colonne TGT\_COL2.
3. Utilisez l'expression TIMESTAMP(CHAR(:SRC\_COL3 ) CONCAT CHAR(:SRC\_COL4 )) pour calculer les valeurs de colonne et pour effectuer le mappage permettant d'obtenir TGT\_COL3. Cette expression de colonne fournit les données qui permettent d'effectuer un mappage avec la colonne d'horodatage de la base de données cible.
4. Utilisez l'expression SUBSTR(:SRC\_COL5,1,5 ) pour calculer les valeurs de colonne et mapper le résultat afin de l'appliquer à la colonne cible TGT\_COL2.



---

## Chapitre 9. Utilisation du programme Capture pour la réplication SQL

Cette section traite de la capture pour bases de données DB2. Si vous utilisez la capture par déclencheurs, les déclencheurs sont créés au moment de l'enregistrement et vous n'effectuez pas les opérations décrites dans cette section.

---

### Démarrage du programme Capture (Linux, UNIX, Windows et z/OS)

Démarrez le programme Capture pour commencer à capturer les données de journaux pour bases de données DB2. Si vous utilisez la capture par déclencheurs pour une source relationnelle non DB2, les déclencheurs sont créés au moment de l'enregistrement et vous n'avez pas besoin de démarrer le programme Capture.

#### Avant de commencer

- Configurez les connexions au serveur source et au serveur de contrôle Capture.
- Assurez-vous que vous disposez des autorisations d'accès appropriées.
- Créez des tables de contrôle pour le schéma Capture approprié.
- Définissez des enregistrements.
- Configurez les programmes Capture et Apply.

#### A propos de cette tâche

**Remarque :** Le programme Capture ne capture pas de modifications apportées par des utilitaires DB2, car ceux-ci ne consignent pas les modifications de façon visible pour le programme Capture.

Lorsque vous démarrez le programme Capture, vous pouvez également sélectionner des paramètres de démarrage.

Après le démarrage du programme Capture, ce dernier peut ne pas commencer immédiatement la capture des données. Il ne commence à capturer les données qu'une fois que le programme Apply lui a signalé qu'il avait entièrement régénéré une table cible. Ensuite, le programme Capture commence à capturer les modifications du journal pour une table source en particulier.

#### Procédure

Pour démarrer le programme Capture sous Linux, UNIX, Windows et z/OS, utilisez l'une des méthodes suivantes :

Méthode	Description
Centre de réplication	Utilisez la fenêtre Démarrer Capture. Pour ouvrir cette fenêtre, affichez la branche <b>Opérations</b> de l'arborescence des objets et cliquez sur le dossier <b>Serveurs de contrôle Capture</b> ; ensuite, cliquez avec le bouton droit de la souris dans le panneau de contenu sur le serveur de contrôle Capture sur lequel réside le programme Capture à démarrer. Sélectionnez <b>Démarrer Capture</b> .

Méthode	Description
<p><b>z/OS</b></p> <p>Linux UNIX Windows</p> <p>Commande système <b>asncap</b></p>	<p>Utilisez cette commande pour démarrer le programme Capture et éventuellement spécifier des paramètres de démarrage.</p>
<p><b>z/OS</b></p> <p>Console ou TSO z/OS</p>	<p>Sur z/OS, vous pouvez démarrer le programme 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 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 Capture aux paramètres PARM=. Vous pouvez indiquer uniquement les paramètres 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><b>Exemple :</b></p> <pre> /* asterisk indicates a comment line // CAP EXEC PGM=ASNCAP,PARMS='LE/Capture parameters' /* Parameters can be any or no LE parameters and any or /* no Capture parameters //SYSIN DD * /* additional Capture parameters, one or more /* parameters on each line   CAPTURE_SERVER=DSN!! CAPTURE_SCHEMA=CAPCAT   DEBUG=Y LOGSTDOU=N </pre>
<p><b>Windows</b></p> <p>Services Windows</p>	<p>Vous pouvez créer un service de réplication DB2 sur les systèmes d'exploitation Windows pour démarrer le programme Capture automatiquement quand le système est démarré.</p>

Pour vérifier si un programme Capture a démarré, appliquez l'une des méthodes suivantes :

- 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.
- Examinez le fichier journal de diagnostic Capture (*serveur\_capture.schéma\_capture.CAP.log* sous z/OS et *instancedb2.serveur\_capture.schéma\_capture.CAP.log* sous Linux, UNIX, et Windows) pour voir si un message indique que le programme est en train de capturer les modifications. Par exemple :

```

ASN0104I Change capture has been started for the source
table "REGRESS.TABLE1" for changes found in the log beginning
with log sequence number "0000:0275:6048".

```
- Vérifiez la table IBMSNAP\_CAPTRACE pour voir si un message indique que le programme capture actuellement les modifications.
- Utilisez la fenêtre Messages 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 Capture contenant le programme Capture dont vous souhaitez afficher les messages et sélectionnez **Rapports > Messages Capture**.

- Utilisez la fenêtre Vérifier le statut dans le Centre de réplication ou la commande **asnccmd status** pour afficher le statut de toutes les unités d'exécution Capture. Pour ouvrir la fenêtre, cliquez avec le bouton droit de la souris sur le serveur Capture sur lequel se trouve le programme Capture que vous souhaitez vérifier et sélectionnez **Vérifier le statut**.

---

## Démarrage du programme Capture à partir d'un emplacement connu dans le journal DB2

Vous pouvez indiquer au programme Capture de relire le journal de reprise DB2 à partir d'un emplacement connu et de retraiter les enregistrements de journal ayant déjà été capturés et appliqués.

### A propos de cette tâche

**Important :** Cette procédure ne doit être utilisée que lorsque la table cible est une copie utilisateur.

### Procédure

1. Arrêtez les programmes Capture et Apply.
2. Définissez les valeurs RETENTION\_LIMIT et LAG\_LIMIT Capture sur leur valeur maximale, comme indiqué dans l'instruction SQL suivante :
3. Si les valeurs SYNCHPOINT des tables IBMSNAP\_UOW, CD, IBMSNAP\_REGISTER et IBMSNAP\_PRUNCNTL sont supérieures à la valeur LSN à partir de laquelle vous voulez démarrer Capture, utilisez SQL pour définir la valeur sur l'emplacement à partir duquel vous voulez commencer à recapturer des transactions. Dans l'exemple suivant, 00000006F5638E60000 est le numéro de séquence du journal et 2009-09-05-09.55.43.316970 est l'horodatage défini pour que le programme Capture recommence la lecture du journal.

```
UPDATE ASN.IBMSNAP_CAPPARMS SET RETENTION_LIMIT=99999,LAG_LIMIT=99999;
```

```
UPDATE ASN.IBMSNAP_REGISTER SET CD_OLD_SYNCHPOINT=x'00000006F5638E600000',  
CD_NEW_SYNCHPOINT=x'00000006F5638E600000',  
CCD_OLD_SYNCHPOINT=x'00000006F5638E600000'  
WHERE GLOBAL_RECORD='N';
```

```
UPDATE ASN.IBMSNAP_SUBS_SET SET LASTRUN=TIMESTAMP  
( '2009-09-05-09.55.43.316970' ),  
LASTSUCCESS=TIMESTAMP( '2009-05-05-09.55.43.316970' ),  
SYNCHPOINT=x'00000006F5638E600000', SYNCHTIME=TIMESTAMP  
( '2009-05-05-09.55.43.316970' )  
WHERE WHOS_ON_FIRST='S' AND SET_NAME='BACK1';
```

```
UPDATE ASN.IBMSNAP_PRUNCNTL SET SYNCHPOINT =x'00000006F5638E600000',  
SYNCHTIME=TIMESTAMP('2009-05-05-09.55.43.316970');
```

```
UPDATE ASN.IBMSNAP_PRUNE_SET SET SYNCHPOINT =x'00000006F5638E600000',  
SYNCHTIME=TIMESTAMP('2009-05-05-09.55.43.316970');
```

```
DELETE FROM ASN.IBMSNAP_UOW;
```

```
INSERT INTO ASN.IBMSNAP_RESTART (MAX_COMMITSEQ, MIN_INFLIGHTSEQ,
```

```

MAX_COMMIT_TIME,CURR_COMMIT_TIME,CAPTURE_FIRST_SEQ)
values (,x'00000006F5638E600000',
'2009-05-05-09.55.43.316970','2009-05-05-09.55.43.316970',
x'00000006F5638E600000');

```

- Démarrez le programme Capture en mode WARMNS et démarrez le programme Apply avec vos paramètres de démarrage normaux.

---

## Démarrage du programme Capture (System i)

Démarrez le programme Capture pour commencer à capturer des données depuis le journal.

### Avant de commencer

Avant de démarrer le programme Capture, veillez à ce que les conditions prérequis suivantes soient satisfaites :

- Vous possédez l'autorisation appropriée.
- Les tables de contrôle sont créées pour le schéma de Capture approprié et les enregistrements sont définis.
- Les programmes de réplication sont configurés si le programme Capture lit un journal distant.

### A propos de cette tâche

Après le démarrage du programme Capture, ce dernier peut ne pas commencer immédiatement la capture des données. Il ne commence à capturer les données qu'une fois que le programme Apply lui a signalé qu'il avait commencé à capturer les modifications dans l'historique pour une table source donnée.

### Procédure

Pour démarrer le programme Capture sur System i, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Commande système STRDPRCAP (System i)</b>	Utilisez la commande Start DPR Capture (STRDPRCAP) pour commencer à capturer les modifications.
<b>Centre de réplication</b>	Utilisez la fenêtre Démarrer Capture. Pour ouvrir cette fenêtre, affichez la branche <b>Opérations</b> de l'arborescence des objets et cliquez sur le dossier <b>Serveurs de contrôle de Capture</b> ; ensuite, cliquez avec le bouton droit de la souris dans le panneau de contenu sur le serveur de contrôle Capture sur lequel réside le programme Capture à démarrer. Sélectionnez <b>Démarrer Capture</b> .

---

## Paramètres d'exploitation par défaut du programme Capture

Lorsque vous créez les tables de contrôle Capture, les valeurs par défaut des paramètres d'exploitation Capture sont enregistrées dans la table IBMSNAP\_CAPPARMS.

Les valeurs par défaut sont indiquées dans le tableau 7, à la page 121 et dans le tableau 8, à la page 122.

Tableau 7. Valeurs par défaut des paramètres d'exploitation Capture (Linux, UNIX, Windows, z/OS)

Paramètre d'exploitation	Valeur par défaut	Nom de colonne dans la table IBMSNAP_CAPPARMS
<b>capture_server</b>	DB2DBDFT <sup>1</sup>	non disponible
<b>capture_schema</b>	ASN <sup>2</sup>	non disponible
<b>add_partition</b>	n <sup>4</sup>	non disponible
<b>asynchlogrd</b>	n <sup>4</sup>	non disponible
<b>retention_limit</b>	10080 minutes	RETENTION_LIMIT
<b>lag_limit</b>	10080 minutes	LAG_LIMIT
<b>commit_interval</b>	30 secondes	COMMIT_INTERVAL
<b>prune_interval</b>	300 secondes	PRUNE_INTERVAL
<b>trace_limit</b>	10080 minutes	TRACE_LIMIT
<b>monitor_limit</b>	10080 minutes	MONITOR_LIMIT
<b>monitor_interval</b>	300 secondes	MONITOR_INTERVAL
<b>memory_limit</b>	32 Mo	MEMORY_LIMIT
<b>autoprune</b>	y <sup>3</sup>	AUTOPRUNE
<b>term</b>	y <sup>3</sup>	TERM
<b>autostop</b>	n <sup>4</sup>	AUTOSTOP
<b>caf</b>	n <sup>4</sup>	non disponible
<b>logread_prefetch</b>	Y <sup>3</sup> pour les bases de données partitionnées ; N <sup>4</sup> pour les bases de données non partitionnées	non disponible
<b>logreuse</b>	n <sup>4</sup>	LOGREUSE
<b>logstdout</b>	n <sup>4</sup>	LOGSTDOUT
<b>sleep_interval</b>	5 secondes	SLEEP
<b>capture_path</b>	Répertoire où Capture a été démarré <sup>5</sup>	CAPTURE_PATH
<b>startmode</b>	warmsi <sup>6</sup>	STARTMODE

**Remarque :**

1. Le serveur de contrôle Capture correspond à la valeur de la variable d'environnement DB2DBDFT pour Windows, Linux et UNIX, si cette variable est spécifiée. Il n'existe pas de valeur par défaut pour z/OS.
2. Vous ne pouvez pas modifier la valeur par défaut du schéma Capture. Pour utiliser un autre schéma Capture, utilisez le paramètre de démarrage **capture\_schema**.
3. Yes
4. No
5. Si Capture démarre en tant que service Windows, son emplacement de capture est \sqllib\bin.
6. Le programme Capture démarre à chaud. Il bascule uniquement sur le mode de démarrage à froid s'il est exécuté pour la première fois.

System i

Tableau 8. Configuration par défaut des paramètres d'exploitation Capture (System i)

Paramètre d'exploitation	Valeur par défaut	Nom de colonne dans la table IBMSNAP_CAPPARMS
<b>CAPCTLLIB</b>	ASN <sup>1</sup>	non disponible
<b>JOBID</b>	*LIBL/QZSNDPR	non disponible
<b>JRN</b>	*ALL	non disponible
<b>RETAIN</b>	10080 minutes	RETENTION_LIMIT
<b>LAG</b>	10080 minutes	LAG_LIMIT
<b>FRCFRQ</b>	30 secondes	COMMIT_INTERVAL
<b>CLNUPITV</b>	*IMMED <sup>2</sup>	non disponible
<b>CLNUPITV</b>	86400 secondes <sup>2</sup>	PRUNE_INTERVAL
<b>CLNUPITV</b>	*IMMED <sup>2</sup>	non disponible
<b>TRCLMT</b>	10080 minutes	TRACE_LIMIT
<b>MONLMT</b>	10080 minutes	MONITOR_LIMIT
<b>MONITV</b>	300 secondes	MONITOR_INTERVAL
<b>MEMLMT</b>	32 Mo	MEMORY_LIMIT
<b>WAIT</b>	120 secondes	non disponible
<b>RESTART</b>	*YES <sup>3</sup>	non disponible

**Remarque :**

1. Vous ne pouvez pas modifier la valeur par défaut du schéma Capture. Pour utiliser un autre schéma Capture, affectez une valeur au paramètre **CAPCTLLIB** lorsque vous lancez le programme Capture. Les valeurs par défaut de la plupart des autres paramètres d'exploitation sont conservées dans la table IBMSNAP\_CAPPARMS.
2. Le paramètre **CLNUPITV** possède deux sous-paramètres. Par défaut, le programme Capture effectue un élagage peu après son démarrage, puis à chaque intervalle d'élagage (toutes les 24 heures, par défaut).
3. Par défaut, le programme Capture démarre à chaud.

## Description des paramètres d'exploitation Capture

Lorsque vous lancez le programme Capture, vous pouvez sélectionner des paramètres de démarrage. Voici les paramètres de démarrage ; pour chacun d'entre eux, des recommandations de choix de valeur sont mentionnées.

Tous les paramètres s'appliquent à z/OS, Linux, UNIX, et Windows, à moins que cela ne soit indiqué différemment.

- «Paramètre add\_partition (Linux, UNIX, Windows)», à la page 123
- «asynchlogrd», à la page 123
- «autoprune», à la page 123
- «autostop», à la page 124
- «caf (z/OS)», à la page 124
- «capture\_path», à la page 125
- «capture\_schema», à la page 125
- «capture\_server», à la page 126
- «commit\_interval», à la page 126
- «hs (z/OS)», à la page 127

- «ignore\_transid», à la page 127
- «lag\_limit», à la page 127
- «logreuse», à la page 128
- «logstdout», à la page 128
- «memory\_limit», à la page 129
- «monitor\_interval», à la page 129
- «prune\_interval», à la page 130
- «retention\_limit», à la page 130
- «sleep\_interval», à la page 131
- «startmode», à la page 131
- «term», à la page 132
- «trace\_limit», à la page 132

Linux UNIX Windows

## Paramètre **add\_partition** (Linux, UNIX, Windows)

Valeur par défaut : **add\_partition=n**

Le paramètre **add\_partition** indique si le programme Capture doit commencer à lire le fichier journal des partitions qui ont été ajoutées depuis le redémarrage du programme Capture.

Spécifiez **add\_partition=y** pour que le programme Capture lise les fichiers journaux. Sur chaque nouvelle partition, lorsque le programme Capture est démarré à chaud, il lit le fichier journal en commençant par le premier LSN utilisé par DB2 après l'émission de la première instruction CONNECT de base de données pour l'instance DB2.

## **asynchlogrd**

Valeur par défaut : **asynchlogrd=n**

Le paramètre **asynchlogrd** indique le programme Capture doit utiliser la même unité d'exécution dédiée pour capturer les transactions à partir du journal de reprise DB2. L'unité d'exécution du programme de lecture des transactions effectue une prélecture des transactions validées dans une mémoire tampon, d'où une autre unité d'exécution récupère les transactions et les traite en instructions SQL pour les insérer dans la table CD. Ce mode asynchrone permet d'améliorer les performances du programme Capture dans tous les environnements grâce à des avantages particuliers pour les bases de données partitionnées et le partage de données sous z/OS.

Sur les systèmes dont les niveaux d'activité sont très élevés, cette prélecture peut entraîner une utilisation plus importante de la mémoire. Ajustez le paramètre **memory\_limit** en conséquence. Si le volume de vos modifications est faible, vous pouvez préférer utiliser la valeur par défaut N pour réduire la consommation d'unité centrale.

## **autoprune**

Valeur par défaut : **autoprune=y**

Le paramètre **autoprune** indique si le programme Capture doit élaguer automatiquement certaines tables de contrôle. Par défaut, avec le paramétrage **autoprune=y**, le programme Capture élague automatiquement les lignes des tables CD et UOW, ainsi que les tables IBMSNAP\_CAPTRACE, IBMSNAP\_CAPMON et IBMSNAP\_SIGNAL. Si vous spécifiez **autoprune=n**, vous devez utiliser la commande `prune` pour élaguer ces tables.

Si vous lancez Capture avec la fonction d'élagage automatique activée, vous devez définir l'intervalle d'élagage afin d'utiliser la fréquence d'élagage la plus adaptée à votre environnement de réplication. Le programme Capture utilise les paramètres suivants pour déterminer les lignes suffisamment anciennes pour être élaguées :

- **retention\_limit** pour les tables CD, UOW et les tables de signaux
- **monitor\_limit** pour les tables de contrôle
- **trace\_limit** pour la table de trace Capture

## autostop

Valeur par défaut : **autostop=n**

Le paramètre **autostop** contrôle si le programme Capture se poursuit ou se termine lorsqu'il atteint la fin du journal.

Par défaut (**autostop=n**), le programme Capture ne se termine pas après l'extraction des transactions.

Utilisez l'option **autostop=y** si vous effectuez une réplication en environnement mobile ou connecté occasionnellement. Le paramètre **autostop** permet de garantir que le programme Capture extrait toutes les transactions admissibles, puis s'arrête lorsqu'il atteint la fin du journal. Pour extraire d'autres transactions, vous devez redémarrer le programme Capture. Vous pouvez également utiliser l'option **autostop=y** en environnement de test.

**Recommandation** : dans la plupart des cas, vous ne devez pas utiliser **autostop=y**, car cela ajoute une grande quantité de temps système à l'administration de la réplication (par exemple, cela vous oblige à redémarrer continuellement le programme Capture).

z/OS

## caf (z/OS)

Valeur par défaut : **n**

L'option par défaut est **caf =n**. Vous pouvez remplacer cette valeur par défaut et indiquer au programme Capture d'utiliser la fonction de connexion d'appel CAF en définissant l'option **caf =y**. L'option **caf =y** indique que le programme de réplication remplace la connexion RRS par défaut et s'exécute avec la connexion CAF.

Si RRS n'est pas disponible, un message vous l'indiquera et le programme de réplication utilisera 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.



## capture\_path

Le chemin d'accès à Capture représente le répertoire dans lequel le programme Capture stocke ses fichiers de travail et son fichier journal. Par défaut, le chemin d'accès à Capture est le répertoire dans lequel vous lancez le programme.

### z/OS

Le programme Capture est une application POSIX ; par conséquent, le chemin d'accès à Capture par défaut dépend du mode de démarrage du programme :

#### Invite de commande USS

Le répertoire dans lequel vous avez démarré le programme.

#### Tâche démarrée ou via JCL

Répertoire de base du système de fichiers USS de l'ID utilisateur associé à la tâche ou au travail démarré.

Vous pouvez indiquer soit un nom de chemin, soit un qualificatif de haut niveau (HLQ), tel que //CAPV9. Lorsque vous utilisez un HLQ, des fichiers séquentiels sont créés conformément aux conventions de dénomination z/OS pour ce type de noms 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 à un emplacement explicitement nommé (le qualificatif HLQ est dans ce cas concaténé en première partie du nom de fichier). Par exemple : *sysadm.CAPV8.nomfichier*. L'utilisation d'un qualificatif HLQ peut être utile si vous voulez que les fichiers LOADMSG et les fichiers journaux Capture soit gérés par le système(SMS).

Si vous souhaitez que la tâche démarrée de Capture ait un droit en écriture sur un ensemble de données .log avec un ID utilisateur différent de l'ID à l'origine de l'exécution de la tâche (par exemple TSOUSER), vous devez indiquer une apostrophe (') comme caractère d'échappement lors de l'utilisation du format SYSIN pour les paramètres d'entrée dans la tâche démarrée. Par exemple, si vous voulez utiliser le qualificatif de haut niveau JOESMITH, l'ID utilisateur TSOUSER qui exécute le programme Capture doit alors disposer des droits RACF en écriture sur les ensembles de données avec le qualificatif de haut niveau JOESMITH, comme illustré :

```
//SYSIN DD *  
  CAPTURE_PATH=/'JOESMITH  
/*
```

### Linux UNIX Windows

Vous pouvez modifier le chemin d'accès à Capture pour spécifier l'emplacement à utiliser par Capture pour le stockage de ses fichiers. Vous pouvez spécifier un nom (/home/db2inst/fichiers\_Capture, par exemple). Si vous lancez le programme Capture en tant que service Windows, le programme Capture est démarré par défaut dans le répertoire \sql\lib\bin.

## capture\_schema

Valeur par défaut : **capture\_schema=ASN**

Le paramètre **capture\_schema** permet d'identifier le programme Capture à démarrer. Par défaut, le schéma Capture est ASN.

Si vous avez déjà créé un autre schéma, vous pouvez démarrer le programme Capture en spécifiant ce schéma (pour cela, utilisez le paramètre **capture\_schema**).

Vous pouvez utiliser plusieurs schémas Capture dans les cas suivants :

### Indépendance d'applications

Vous pouvez créer plusieurs schémas Capture afin d'utiliser un programme Capture pour l'application A et un autre programme Capture pour l'application B. Chaque programme Capture utilise ses propres tables de contrôle. Si l'un des programmes Capture est arrêté, une seule application s'en trouve affectée. L'autre application n'est pas affectée, car elle est exécutée par un autre programme Capture.

### Besoins différents en matière d'applications

Vous pouvez créer plusieurs schémas Capture si différentes applications utilisent les mêmes tables source, tout en ayant des besoins différents en matière de données. Par exemple, une application de gestion de la paie a besoin d'utiliser des données sensibles sur les employés, alors qu'un registre interne d'employés n'a pas ces mêmes exigences. Vous pouvez enregistrer les informations confidentielles dans l'un des schémas Capture, mais non dans l'autre. De la même façon, vous pouvez enregistrer une table plusieurs fois si certaines applications exigent que le programme Capture ait un comportement différent. Par exemple, certaines applications ont peut-être besoin que le programme Capture enregistre les mises à jour sous forme de couples de suppressions/insertions.

### Isolement des incidents d'enregistrement

Si un incident se produit lors d'un enregistrement, vous pouvez créer un autre schéma Capture et y transférer les enregistrements de travail. De cette façon, vous pouvez déboguer l'enregistrement dans le schéma d'origine et exécuter les enregistrements non affectés à l'aide de l'autre schéma.

## capture\_server

**z/OS**

Valeur par défaut : `capture_server=None`

**Linux UNIX Windows**

Valeur par défaut: `capture_server=valeur` de la variable d'environnement `DB2DBDFT`, si elle est définie

Le paramètre `capture_server` spécifie le serveur de contrôle Capture.

**z/OS**

Vous devez définir le paramètre `capture_server`. Les tables de contrôle Capture se trouve à l'emplacement du nom du sous-système DB2. Vu que le programme Capture lit le journal DB2, il doit s'exécuter sur le même serveur que la base de données source.

**Linux UNIX Windows**

Les tables de contrôle Capture (telles que la table de registres) contiennent les informations d'enregistrement des tables source et se trouvent sur le serveur de contrôle Capture.

## commit\_interval

Valeur par défaut : `commit_interval=30`

Le paramètre `commit_interval` spécifie la fréquence de validation (en secondes), par le programme Capture, des données dans les tables de contrôle Capture (y compris les tables UOW et CD). Par défaut, le programme Capture attend 30 secondes avant de valider les données dans les tables CD et UOW. Des verrous

sont appliqués aux tables mises à jour pendant l'intervalle de validation. Si vous affectez des valeurs plus élevées au paramètre **commit\_interval**, cela diminue l'utilisation d'UC par le programme Capture, mais cela risque d'augmenter le délai de latence des ensembles d'abonnements fréquemment exécutés, car le programme Apply ne peut rechercher que des données validées.

## hs (z/OS)

Valeur par défaut : **hs=n**

Le paramètre **hs** indique, quelque soit le nombre de fichiers auxiliaires créés dans l'hiperspace (espace de données hautes performances), si le programme Capture excède sa limite de mémoire au cours d'une tentative d'écriture d'une ligne en mémoire. Par défaut (**hs=n**), le programme Capture crée le fichier auxiliaire sur disque ou dans les E/S virtuelles (VIO).

**Recommandation** : Allouez suffisamment d'espace mémoire au travail du programme Capture afin d'éviter de créer des fichiers auxiliaires.

## ignore\_transid

Valeur par défaut : aucune

Le paramètre **ignore\_transid=ID\_transaction** indique que le programme Capture ignore la transaction identifiée par l'ID *ID\_transaction*. Les transactions ne sont pas répliquées ni publiées. Vous pouvez utiliser ce paramètre si vous voulez ignorer une transaction volumineuse qu'il n'est pas nécessaire de répliquer. Par exemple, un travail par lots de grande taille. 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* est l'ID de transaction et *mmmm*, 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.

**Linux UNIX Windows**

mmmm:0000:xxxx:xxxx:xxxx

où *xxxx:xxxx:xxxx* est l'ID transaction et *mmmm* 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.

## lag\_limit

Valeur par défaut : **lag\_limit=10 080**

Le paramètre **lag\_limit** représente le nombre de minutes que le programme Capture peut consacrer au traitement des enregistrements du journal DB2.

Par défaut, si les enregistrements de journal sont plus anciens que 10 080 minutes (soit sept jours), le programme Capture ne démarre que si vous indiquez une

valeur pour le paramètre Capture **startmode**, permettant ainsi au programme Capture de passer en mode de démarrage à froid.

Si le programme Capture ne démarre toujours pas car la limite est atteinte, vous devez déterminer la cause du retard de lecture du journal par le programme Capture. Si vous vous trouvez en environnement de test, dans lequel vous ne disposez pas de l'utilisation pratique de ce paramètre, vous devez peut-être définir une valeur plus élevée et tenter de nouveau de démarrer le programme Capture. À l'inverse, si votre table source contient très peu de données en environnement de test, vous devez peut-être utiliser un démarrage à froid et non une régénération intégrale des données au sein de toutes les tables cible.

## logreuse

**Valeur par défaut : logreuse=n**

Le programme Capture conserve les informations opérationnelles dans un fichier journal.

**z/OS** Le nom du fichier journal ne contient pas le nom d'instance de DB2. (Par exemple : SRCDB1.ASN.CAP.log.) Ce fichier se trouve dans le répertoire indiqué par le paramètre **capture\_path**. Si le paramètre **capture\_path** est spécifié en tant que HLQ (High Level Qualifier), les conventions de dénomination de fichier séquentiel z/OS s'appliquent : par conséquent, le nom spécifié pour **capture\_schema** utilisé pour l'élaboration du nom de fichier journal est tronqué aux 8 premiers caractères du nom.

**Linux UNIX Windows** Le nom du fichier journal est *instancedb2.serveur\_capture.schema\_capture.CAP.log*. (Par exemple : DB2INST.SRCDB1.ASN.CAP.log.)

Par défaut (**logreuse=n**), le programme Capture ajoute des messages au fichier journal, même après le redémarrage du programme Capture. Si vous souhaitez disposer de l'historique des messages, conservez les valeurs par défaut. Dans les cas suivants, vous souhaitez peut-être que le programme Capture supprime le journal, puis le crée de nouveau lorsqu'il redémarre (**logreuse=y**) :

- La taille du journal devient volumineuse et vous souhaitez nettoyer le journal.
- Vous n'avez pas besoin de l'historique qui se trouve dans le journal.
- Vous souhaitez économiser de l'espace.

## logstdout

**Valeur par défaut : logstdout=n**

Le paramètre **logstdout** n'est disponible que si vous utilisez la commande **asncap** ; il n'est pas disponible dans le Centre de réplication.

Par défaut, le programme Capture envoie des messages d'avertissement et d'information au fichier journal uniquement. Vous pouvez choisir d'envoyer ces messages à la sortie standard (**logstdout=y**) en cas d'identification d'incident ou si vous surveillez la fréquence de fonctionnement du programme Capture en environnement de test.

## memory\_limit

Valeur par défaut : `memory_limit=32`

Le paramètre `memory_limit` spécifie la quantité de mémoire (exprimée en mégaoctets) que le programme Capture peut utiliser.

Par défaut, le programme Capture utilise 32 mégaoctets de mémoire pour le stockage des informations de transactions, avant de les déverser dans un fichier situé dans le répertoire `capture_path`. Vous pouvez modifier la limite de mémoire en fonction de vos besoins en matière de performances. Si vous affectez une valeur de limite de mémoire plus élevée, cela peut accroître les performances de Capture, mais risque de diminuer la quantité de mémoire disponible pour d'autres utilisations sur votre système. Si vous affectez une valeur de limite de mémoire plus faible, cela libère de la mémoire en vue d'autres utilisations. Toutefois, si vous utilisez une limite de mémoire trop basse et que le programme Capture déverse les données dans un fichier, vous utiliserez davantage d'espace système et cela induira plus d'E/S susceptibles de ralentir votre système.

Vous pouvez surveiller la limite de mémoire via l'utilisation du moniteur d'alertes de réplication. Vous pouvez également utiliser les données contenues dans la table CAPMON pour déterminer le nombre de transactions de système source déversées sur le disque en raison des restrictions de mémoire. Effectuez le total des valeurs présentes dans la colonne TRANS\_SPILLED de la table CAPMON.

## monitor\_interval

Valeur par défaut : `monitor_interval=300`

Le paramètre `monitor_interval` indique la fréquence à laquelle le programme Capture enregistre les informations dans la table IBMSNAP\_CAPMON.

Par défaut, le programme Capture insère des lignes dans la table de contrôle Capture toutes les 300 secondes (5 minutes). Ce paramètre opérationnel fonctionne avec l'intervalle de validation. Si vous désirez un suivi plus granulaire des données, utilisez un intervalle d'interception plus proche de l'intervalle de validation.

## monitor\_limit

Valeur par défaut: `monitor_limit=10080`

Le paramètre `monitor_limit` indique le degré d'ancienneté que les lignes de la table de contrôle doivent présenter avant de pouvoir être élaguées.

Par défaut, les lignes de la table IBMSNAP\_CAPMON antérieures à 10 080 minutes (sept jours) sont élaguées. La table IBMSNAP\_CAPMON contient les statistiques opérationnelles du programme Capture. Utilisez la limite d'interception par défaut si vous avez besoin de moins d'une semaine de statistiques. Si vous suivez fréquemment l'activité statistique, vous n'avez sans doute pas besoin de conserver une semaine de statistiques et vous pouvez définir une limite d'interception inférieure, afin que la table d'interception Capture soit élaguée plus souvent et que les anciennes statistiques soient supprimées. Si vous souhaitez utiliser les statistiques à des fins d'analyse d'historique et que vous souhaitez examiner plus d'une semaine de statistiques, augmentez la valeur de ce paramètre.

## prune\_interval

Valeur par défaut : **prune\_interval=300**

Le paramètre **prune\_interval** indique la fréquence selon laquelle le programme Capture tente d'élaguer les anciennes lignes de certaines de ses tables de contrôle. Ce paramètre n'est valide que si **autoprun=y**.

Par défaut, le programme Capture effectue l'élagage des tables CD et UOW toutes les 300 secondes (cinq minutes). Si ces tables ne sont pas assez souvent élaguées, elles peuvent excéder leurs limites d'espace table et provoquer alors l'arrêt du programme Capture. Inversement, si elles sont élaguées trop souvent ou au cours de périodes de pointe, cette opération d'élagage peut faire interférence avec d'autres applications s'exécutant sur le même système. Vous pouvez définir la fréquence d'élagage optimale pour votre environnement de réplication. Les performances sont généralement plus élevées lorsque la taille des tables est limitée.

Avant de diminuer la valeur d'intervalle d'élagage, vérifiez que les données sont appliquées fréquemment, ce qui permet de les élaguer. Si le programme Apply n'applique pas les données fréquemment, il est inutile de définir un intervalle d'élagage plus faible, car le programme Apply doit répliquer les données sur toutes les cibles avant de pouvoir élaguer les tables CD et UOW.

L'intervalle d'élagage détermine la fréquence selon laquelle le programme Capture tente d'élaguer les tables. Il fonctionne en conjonction avec les paramètres suivants, qui déterminent à partir de quel moment les données sont suffisamment anciennes pour être élaguées : **trace\_limit**, **monitor\_limit**, **retention\_limit**. Par exemple, si le paramètre **prune\_interval** porte la valeur 300 secondes et que le paramètre **trace\_limit** porte la valeur 10080 secondes, le programme Capture tentera d'effectuer un élagage toutes les 300 secondes. S'il détecte dans la table de trace des lignes de plus de 10 080 minutes (7 jours), il les élague.

## retention\_limit

Valeur par défaut : **retention\_limit=10080**

Le paramètre **retention\_limit** détermine la durée de conservation des anciennes données dans les tables CD, UOW et IBMSNAP\_SIGNAL avant d'être admissibles pour l'élagage de limite de conservation.

Si le processus d'élagage normal est empêché pour cause d'ensembles d'abonnements désactivés ou exécutés peu souvent, les données sont conservées dans les tables CD et UOW pendant de longues périodes. Lorsque ces données sont plus anciennes que l'horodatage DB2 actuel moins la valeur de durée de conservation, le processus d'élagage de durée de conservation supprime ces données des tables. Si vous exécutez vos ensembles d'abonnements très peu souvent ou que vous arrêtez les programmes Apply, la taille de vos tables CD et UOW peut croître énormément ; ces tables deviennent alors admissibles pour l'élagage de durée de conservation.

Vos tables cible doivent être actualisées afin d'être synchronisées avec la source, si certaines lignes élaguées sont candidates à la réplication, mais que pour des raisons diverses, elles n'ont pas encore été appliquées à la table cible. Vous pouvez éviter une régénération intégrale en utilisant des durées de conservation plus élevées ; toutefois, dans ce cas, la taille de vos tables CD et UOW augmente et cela utilise de l'espace système.

Si vous utilisez la réplication bidirectionnelle, l'élagage de durée de conservation permet de s'assurer que les transactions rejetées ont bien été supprimées. Les transactions rejetées résultent de l'utilisation de la détection de conflit avec les tables cible réplique et de la détection de transactions en conflit. Les lignes des tables CD et UOW appartenant à ces transactions rejetées ne sont pas répliquées et sont élaguées une fois la limite de durée de conservation atteinte. Une régénération intégrale n'est pas nécessaire si toutes les anciennes lignes supprimées appartenaient à des transactions rejetées.

L'élagage de durée de conservation garantit également que les informations sur les signaux devenues inutiles sont supprimées de la table IBMSNAP\_SIGNAL.

## **sleep\_interval**

**Valeur par défaut : sleep\_interval=5**

L'intervalle d'inactivité représente le nombre de secondes pendant lequel le programme Capture attend de pouvoir lire de nouveau le journal une fois qu'il a atteint la fin du journal et que la mémoire tampon est vide. Pour le partage des données sous z/OS, l'intervalle d'inactivité représente le nombre de secondes pendant lequel le programme Capture reste inactif une fois que la mémoire tampon affiche un remplissage inférieur à la moitié.

Par défaut, le programme Capture reste inactif pendant 5 secondes. Modifiez l'intervalle d'inactivité si vous souhaitez diminuer le temps système utilisé par le programme Capture pour la lecture du journal. Si vous utilisez un intervalle d'inactivité moins élevé, cela signifie que les risques de retard sont moindres. Si vous utilisez un intervalle d'inactivité plus long, vous pouvez réaliser une économie d'UC importante sur un système n'étant pas mis à jour fréquemment.

## **startmode**

**Valeur par défaut : startmode=warmsi**

Vous pouvez démarrer le programme Capture à l'aide de l'un des modes de démarrage suivants :

**warmsi (démarrage à chaud, bascule la toute première fois sur un démarrage à froid)** Le programme Capture démarre à chaud (sauf si vous démarrez le programme Capture pour la première fois) ; ensuite, il passe en mode de démarrage à froid. Utilisez ce mode de démarrage si vous souhaitez vous assurer que le démarrage à froid ne se produit que lors du démarrage initial du programme Capture.

**warmns (démarrage à chaud , ne bascule jamais sur un démarrage à froid)** Le programme Capture démarre à chaud. S'il ne peut pas être démarré à chaud, il ne passe pas à un démarrage à froid. Lorsque vous utilisez **warmns** dans votre environnement de réplication, vous pouvez corriger des problèmes (comme une indisponibilité des bases de données ou des espaces table) empêchant le démarrage à chaud. Utilisez ce mode de démarrage pour empêcher un démarrage à froid à l'improviste. Lorsque le programme Capture démarre à chaud, il reprend le traitement là où il s'était arrêté. Si des erreurs se produisent après le démarrage du programme Capture, le programme s'arrête sans modifier aucune table.

**Conseil** : vous ne pouvez pas utiliser **warmns** pour démarrer le programme Capture pour la première fois, car il n'existe pas d'informations sur le

démarrage à chaud lors du démarrage initial du programme Capture. Utilisez le mode de démarrage **cold** lors du démarrage initial du programme Capture, puis utilisez le mode de démarrage **warmns**. Si vous ne souhaitez pas effectuer de bascules entre les modes de démarrage, vous pouvez choisir d'utiliser de préférence **warmsi**.

**cold** Pendant le démarrage à froid, le programme Capture supprime toutes les lignes de ses tables CD et de sa table UOW, au cours de la phase d'initialisation. Tous les ensembles d'abonnements associés à ces sources de réplication sont totalement actualisés au cours du cycle Apply suivant (c'est-à-dire que toutes les données sont copiées des tables source vers les tables cible). Si le programme Capture tente d'effectuer un démarrage à froid, mais que l'option de régénération intégrale a été désactivée, le programme Capture démarre ; toutefois, le programme Apply échoue et un message d'erreur est émis.

Il est rare que vous demandiez explicitement au programme Capture d'effectuer un démarrage à froid. Le démarrage à froid n'est requis que lorsque le programme Capture démarre pour la première fois ; **warmsi** représente le mode de démarrage recommandé.

**Important** : n'effectuez pas de démarrage à froid du programme Capture si vous souhaitez conserver des historiques précis des données modifiées. Un décalage risque de se produire si le programme Apply ne parvient pas à répliquer les modifications avant l'arrêt du programme Capture. De même, ne définissez pas le mode de démarrage à froid comme valeur par défaut de STARTMODE dans la table IBMSNAP\_CAPPARMS, étant donné que vous souhaitez éviter les démarrages à froid.

## term

**Valeur par défaut** : **term=y**

Le paramètre **term** détermine la façon dont l'état de DB2 affecte le fonctionnement du programme Capture.

Par défaut, le programme Capture s'arrête si DB2 s'arrête.

Utilisez **term=n** si vous voulez que le programme Capture attende que DB2 démarre si DB2 n'est pas actif. Si DB2 est au repos, le programme Capture ne s'arrête pas ; il reste actif mais n'utilise pas la base de données.

## trace\_limit

**Valeur par défaut** : **trace\_limit10080**

Le paramètre **trace\_limit** spécifie le degré d'ancienneté que les lignes doivent présenter dans la table IBMSNAP\_CAPTRACE avant de pouvoir être élaguées.

Lorsque le programme Capture effectue par défaut un élagage, les lignes de la table IBMSNAP\_CAPTRACE sont admissibles pour l'élagage toutes les 10080 minutes (sept jours). La table CAPTRACE contient les informations de trace de contrôle relatives au programme Capture. Toutes les opérations de Capture sont enregistrées dans cette table ; par conséquent, la taille de celle-ci peut augmenter très rapidement en cas d'activité soutenue du programme Capture. Modifiez la limite de trace en fonction de vos besoins en matière d'informations pour audit.



---

## Méthodes de modification des paramètres Capture

Vous pouvez modifier les valeurs enregistrées des paramètres d'exploitation Capture et vous pouvez remplacer temporairement ces valeurs lors du démarrage du programme ou pendant l'exécution de ce dernier.

### Définition de nouvelles valeurs par défaut dans la table IBMSNAP\_CAPPARMS

La table IBMSNAP\_CAPPARMS contient des paramètres que vous pouvez modifier pour contrôler le fonctionnement du programme Capture. Le nom de schéma de la table est le nom de schéma du programme Capture. Une fois la table créée, elle contient les valeurs par défaut applicables au programme Capture. Si la valeur de colonne de la table IBMSNAP\_CAPPARMS n'est pas définie, les valeurs par défaut sont utilisées.

### Indication des valeurs des paramètres au moment du démarrage du programme Capture

Vous pouvez indiquer des valeurs applicables au programme Capture au moment de son démarrage. Les valeurs que vous définissez au cours du démarrage contrôlent le comportement du programme Capture pour la session en cours et remplacent les valeurs des paramètres d'exploitation par défaut, ainsi que les valeurs présentes dans la table de paramètres Capture. Elles ne mettent pas à jour les valeurs de la table de paramètres Capture. Si vous ne modifiez pas la table de paramètres Capture avant de démarrer le programme Capture et que vous ne spécifiez aucun paramètre lors du démarrage du programme Capture, les valeurs par défaut des paramètres d'exploitation sont utilisées.

### Modification des valeurs de paramètres pendant l'exécution du programme Capture

Au cours de l'exécution de Capture, vous pouvez modifier temporairement les paramètres d'exploitation de ce programme. Le programme Capture utilisera alors ces nouvelles valeurs jusqu'à ce que vous les changiez de nouveau, ou jusqu'à ce que vous arrêtiez puis redémarriez le programme Capture. Vous pouvez modifier les paramètres Capture aussi souvent que vous le souhaitez au cours de la session.

Linux UNIX Windows

### Exemple 1

Supposons que vous ne souhaitiez pas utiliser les paramètres par défaut de l'intervalle de validation pour le schéma Capture ASNPROD.

1. Mettez à jour la table de paramètres Capture au niveau du schéma Capture ASNPROD. Affectez la valeur 60 secondes à l'intervalle de validation ; par conséquent, lors du prochain démarrage de ce programme Capture, l'intervalle de validation par défaut sera de 60 secondes.
2. Vous pouvez, le cas échéant, effectuer un réglage des performances, et décider par exemple de démarrer Capture en utilisant un intervalle de validation plus petit. Au lieu de modifier la valeur correspondante dans la table des paramètres Capture, il vous suffit de démarrer le programme Capture en ayant affecté au paramètre d'intervalle de validation la valeur 20 secondes. Tandis que le programme Capture fonctionne avec un intervalle de validation de 20 secondes, vous surveillez ses performances.

```
asncap capture_server=srddb1 capture_schema=asnprod commit_interval=20
```

3. Vous décidez de tenter un intervalle de validation encore plus faible. Au lieu d'arrêter le programme Capture, vous envoyez une demande de modification de paramètres indiquant une valeur d'intervalle de validation de 15 secondes. L'exécution du programme Capture se poursuit, et la validation des données a lieu toutes les 15 secondes.

```
asnccmd capture_server=srcdb1 capture_schema=asnprod chgparms  
commit_interval=15
```

**Important** : le paramètre que vous modifiez doit suivre immédiatement le paramètre **chgparms**.

4. Vous pouvez poursuivre la surveillance des performances et modifier l'intervalle de validation sans arrêter le programme Capture. Lorsque vous avez identifié la valeur répondant à vos besoins, vous pouvez mettre à jour les tables de paramètres Capture (voir la description correspondante à l'étape 1) afin que, lors du prochain démarrage du programme Capture, celui-ci utilise la nouvelle valeur comme intervalle de validation par défaut.

#### System i

### Exemple 2

Supposons que vous ne souhaitiez pas utiliser les paramètres par défaut de l'intervalle de validation pour le schéma Capture ASNPROD.

1. Mettez à jour la table de paramètres Capture au niveau du schéma Capture ASNPROD. Affectez la valeur 90 secondes à l'intervalle de validation ; par conséquent, lors du prochain démarrage de ce programme Capture, l'intervalle de validation par défaut sera de 90 secondes.

```
CHGDPRCAPA CAPCTLLIB(ASNPROD) FRCFRQ(90)
```

2. Vous pouvez, le cas échéant, effectuer un réglage des performances, et décider par exemple de démarrer Capture en utilisant un intervalle de validation plus petit. Au lieu de modifier la valeur correspondante dans la table des paramètres Capture, il vous suffit de démarrer le programme Capture en ayant affecté au paramètre d'intervalle de validation la valeur 45 secondes. Tandis que le programme Capture fonctionne avec un intervalle de validation de 45 secondes, vous surveillez ses performances.

```
STRDPRCAP CAPCTLLIB(ASNPROD) FRCFRQ(45)
```

3. Vous décidez de tenter un intervalle de validation encore plus faible. Au lieu d'arrêter le programme Capture, vous envoyez une demande de modification de paramètres indiquant une valeur d'intervalle de validation de 30 secondes. L'exécution du programme Capture se poursuit, et la validation des données a lieu toutes les 30 secondes. (Remarque : sous System i, vous ne pouvez pas définir de valeur d'intervalle de validation inférieure à 30 secondes.)

```
OVRDPRCAPA CAPCTLLIB(ASNPROD) FRCFRQ(30)
```

4. Lorsque vous avez identifié la valeur répondant à vos besoins, vous pouvez mettre à jour les tables de paramètres Capture (voir la description correspondante à l'étape 1) afin que, lors du prochain démarrage du programme Capture, celui-ci utilise la nouvelle valeur comme intervalle de validation par défaut.

---

## Modification du comportement d'un programme Capture en cours d'exécution

Vous pouvez modifier dynamiquement la valeur d'un ou de plusieurs paramètres d'exploitation d'un programme Capture. Les modifications ne sont pas enregistrées dans la table IBMSNAP\_CAPPARMS, mais sont utilisées jusqu'à ce que vous arrêtez le programme Capture ou que vous entriez de nouvelles valeurs.

### A propos de cette tâche

**z/OS** **Linux UNIX Windows** Vous pouvez modifier les paramètres Capture suivants pendant l'exécution du programme Capture :

- **autoprune**
- **autostop**
- **commit\_interval**
- **lag\_limit**
- **logreuse**
- **logstdout**
- **memory\_limit**
- **monitor\_interval**
- **monitor\_limit**
- **prune\_interval**
- **retention\_limit**
- **sleep\_interval**
- **term**
- **trace\_limit**

**Restriction :** **z/OS** La quantité de mémoire que le programme Capture peut utiliser pour l'élaboration de messages est déterminée lors du démarrage du programme Capture, en fonction de la valeur du paramètre **memory\_limit** et de la taille de la REGION spécifiée en JCL. La valeur de **memory\_limit** n'est pas modifiable si le programme Capture est en cours d'exécution. Pour modifier cette valeur, vous devez d'abord arrêter le programme Capture.

**System i** Vous pouvez remplacer les valeurs des paramètres d'exploitation suivants pour un schéma Capture déterminé :

- **CLNUPITV**
- **FRCFRQ**
- **MEMLMT**
- **MONLMT**
- **MONITV**
- **PRUNE**
- **RETAIN**
- **TRCLMT**

Lorsque vous modifiez ces valeurs, vos modifications ne prennent pas nécessairement effet immédiatement pour tous les paramètres.

## Procédure

Pour modifier le comportement d'un programme Capture en cours d'exécution, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Centre de réplication</b>	Utilisez la fenêtre Modification des paramètres pour l'exécution d'un programme Capture. Cette méthode permet d'afficher les valeurs affectées aux paramètres avant de les modifier. Pour ouvrir cette fenêtre, affichez la branche <b>Opérations</b> de l'arborescence des objets, puis cliquez sur <b>Serveurs de contrôle Capture</b> ; ensuite, cliquez avec le bouton droit sur un serveur de contrôle Capture dans le panneau de contenu et sur <b>Changer les paramètres &gt; Exécution d'un programme Capture</b> .
<b>z/OS</b> <b>Linux UNIX Windows</b> Commande système <b>asncmd chgparms</b>	Cette méthode ne permet pas d'afficher les valeurs actuelles des paramètres.
<b>System i</b> Commande système <b>OVRDPRCAPA</b>	Utilisez la commande Ignorer les attributs DPR Capture (OVRDPRCAPA) pour modifier le comportement d'un programme Capture en cours d'exécution.

---

## Modification des paramètres d'exploitation enregistrés dans la table IBMSNAP\_CAPPARMS

La table IBMSNAP\_CAPPARMS contient les paramètres d'exploitation enregistrés pour le programme Capture. Lorsque vous démarrez le programme Capture, ce dernier utilise les valeurs de cette table, sauf si vous les remplacez temporairement en utilisant les paramètres de démarrage ou pendant le fonctionnement du programme.

### A propos de cette tâche

Une seule ligne est autorisée dans la table IBMSNAP\_CAPPARMS et celle-ci est obligatoire. Si vous souhaitez modifier une ou plusieurs valeurs par défaut, vous pouvez mettre à jour les colonnes au lieu d'insérer des lignes.

Le programme Capture ne lit cette table qu'au cours du démarrage. Si vous modifiez la table de paramètres Capture pendant que ce programme est en cours d'exécution et de réinitialisation, cela ne modifie pas le fonctionnement du programme.

## Procédure

Pour modifier les paramètres enregistrés dans la table IBMSNAP\_CAPPARMS, appliquez l'une des méthodes suivantes :

Méthode	Description
<b>Centre de réplication</b>	Utilisez la fenêtre Modifier les paramètres sauvegardés. Pour ouvrir cette fenêtre, affichez la branche <b>Opérations</b> de l'arborescence des objets, puis cliquez sur <b>Serveurs de contrôle Capture</b> ; ensuite, cliquez avec le bouton droit de la souris sur un serveur de contrôle Capture dans le panneau de contenu et sur <b>Modifier les paramètres &gt; sauvegardés</b> .
<div style="background-color: #800000; color: white; padding: 2px; text-align: center;">System i</div> Commande système <b>CHGDPRCAPA</b>	Utilisez la commande de modification des attributs Capture (CHGDPRCAPA) pour modifier les paramètres d'exploitation globaux utilisés par le programme Capture et stockés dans la table IBMSNAP_CAPPARMS.

Les modifications de paramètres ne sont prises en considération qu'une fois que vous avez arrêté puis redémarré le programme Capture.

## Arrêt du programme Capture

Vous pouvez arrêter le programme Capture pour un schéma Capture en particulier. Lorsque vous arrêtez le programme Capture, il ne capture plus les données de la source.

### A propos de cette tâche

System i

 Si vous choisissez de réorganiser la table UOW ainsi que toutes les tables de modifications des données (CD) qui étaient ouvertes au moment de l'arrêt du programme Capture, cet arrêt prend quelques instants (il n'est pas immédiat).

### Procédure

Pour arrêter le programme Capture, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Centre de réplication</b>	Utilisez la fenêtre Arrêter Capture. Pour ouvrir cette fenêtre, affichez la branche <b>Opérations</b> de l'arborescence des objets, puis cliquez sur <b>Serveurs de contrôle Capture</b> ; ensuite, cliquez avec le bouton droit de la souris sur un serveur de contrôle Capture dans le panneau de contenu et sur <b>Arrêter Capture</b> .
<div style="background-color: #800000; color: white; padding: 2px; text-align: center;">z/OS</div> <div style="background-color: #800000; color: white; padding: 2px; text-align: center;">Linux UNIX Windows</div> Commande système <b>asnccmd stop</b>	Utilisez cette commande pour arrêter Capture.
<div style="background-color: #800000; color: white; padding: 2px; text-align: center;">System i</div> Commande système <b>ENDDPRCAP</b>	Utilisez la commande d'arrêt de Capture (ENDDPRCAP) pour arrêter le programme Capture.

Si vous arrêtez ou interrompez le programme Capture pendant l'élagage, l'élagage est également interrompu. Lorsque vous reprenez ou redémarrez le programme Capture, l'élagage reprend sur la base de la valeur du paramètre **autoprune**.

Il est inutile d'arrêter le programme Capture pour supprimer un enregistrement. Veuillez à désactiver systématiquement un enregistrement avant de le supprimer.

---

## Réinitialisation de Capture

Vous devez réinitialiser le programme Capture si vous modifiez certains attributs des objets enregistrés existants pendant le fonctionnement de ce programme.

### A propos de cette tâche

Par exemple, vous devez réinitialiser le programme Capture si vous modifiez les valeurs CONFLICT\_LEVEL, CHGONLY, RECAPTURE, CHG\_UPD\_TO\_DEL\_INS de la table IBMSNAP\_REGISTER.

Pour le programme Capture sous System i, la réinitialisation est également requise pour commencer la capture des données au niveau d'un journal qui ne faisait pas l'objet de captures auparavant.

### Procédure

Pour réinitialiser le programme Capture, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Centre de réplication</b>	Utilisez la fenêtre Réinitialiser Capture. Pour ouvrir cette fenêtre, affichez la branche <b>Opérations</b> de l'arborescence des objets, puis cliquez sur <b>Serveurs de contrôle Capture</b> ; ensuite, cliquez avec le bouton droit de la souris sur un serveur de contrôle Capture dans le panneau de contenu et sur <b>Réinitialiser Capture</b> .
<b>z/OS</b> <b>Linux UNIX Windows</b> Commande système <b>asncmd reinit</b>	Utilisez cette commande pour réinitialiser Capture.
<b>System i</b> Commande système <b>INZDPRCAP</b>	Utilisez la commande d'initialisation de Capture (INZDPRCAP) pour initialiser le programme Capture.

---

## Interruption du programme Capture (Linux, UNIX, Windows, z/OS)

Vous pouvez interrompre le programme Capture pour libérer des ressources de système d'exploitation au cours de périodes de pointe, sans porter atteinte à l'environnement du programme Capture.

### Avant de commencer

Le programme Capture doit être démarré avec le schéma spécifique correspondant.

### A propos de cette tâche

Vous pouvez également interrompre le programme Capture au lieu de l'arrêter si vous ne souhaitez pas qu'il s'arrête après avoir terminé le travail en cours. Lorsque vous indiquez au programme Capture qu'il doit reprendre, vous n'avez pas besoin de la reprise du temps système Capture.

**Important :** N'interrompez pas le programme Capture avant d'avoir supprimé une source de réplication. Il est préférable de désactiver la source de réplication avant de la supprimer.

### Procédure

Pour interrompre le programme Capture, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Centre de réplication</b>	Utilisez la fenêtre Interrompre Capture. Pour ouvrir cette fenêtre, affichez la branche <b>Opérations</b> de l'arborescence des objets, puis cliquez sur <b>Serveurs de contrôle Capture</b> ; ensuite, cliquez avec le bouton droit de la souris sur un serveur de contrôle Capture dans le panneau de contenu et sur <b>Interrompre Capture</b> .
<b>z/OS</b> <b>Linux UNIX Windows</b> Commande système <b>asncmd suspend</b>	Utilisez cette commande pour interrompre Capture.

Si vous arrêtez ou interrompez le programme Capture pendant l'élagage, l'élagage est également interrompu. Lorsque vous reprenez ou redémarrez le programme Capture, l'élagage reprend sur la base de la valeur du paramètre **autoprune**.

---

## Reprise de Capture (Linux, UNIX, Windows, z/OS)

Vous devez effectuer la reprise d'un programme Capture interrompu si vous souhaitez qu'il recommence à capturer les données.

### Procédure

Pour effectuer la reprise d'un programme Capture interrompu, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Centre de réplication</b>	Utilisez la fenêtre Reprendre Capture. Pour ouvrir cette fenêtre, affichez la branche <b>Opérations</b> de l'arborescence des objets, puis cliquez sur <b>Serveurs de contrôle Capture</b> ; ensuite, cliquez avec le bouton droit de la souris sur un serveur de contrôle Capture dans le panneau de contenu et sur <b>Reprendre Capture</b> .
<b>z/OS</b> <b>Linux UNIX Windows</b> Commande système <b>asncmd resume</b>	Utilisez cette commande pour effectuer la reprise du programme Capture.

Si vous arrêtez ou interrompez le programme Capture pendant l'élagage, l'élagage est également interrompu. Lorsque vous reprenez ou redémarrez le programme Capture, l'élagage reprend sur la base de la valeur du paramètre **autoprune**.





---

## Chapitre 10. Utilisation du programme Apply pour la réplication SQL

Le fonctionnement du programme Apply comprend des tâches telles que démarrer et arrêter et l'utilisation des routines d'exit ASNDONE et ASNLOAD.

---

### Démarrage du programme Apply (Linux, UNIX, Windows, z/OS)

Vous pouvez démarrer une instance du programme Apply pour commencer à appliquer des données à vos cibles.

#### Avant de commencer

Vérifiez que :

- Les connexions sont configurées pour tous les serveurs de réplication nécessaires.
- Vous possédez l'autorisation appropriée.
- Les tables de contrôle contenant les données source et de contrôle pour le qualificatif Apply sont créées.
- Les programmes de réplication sont configurés.
- **z/OS** Vous associez manuellement le programme Apply à tous les serveurs nécessaires.
- **Linux UNIX Windows** Il existe un fichier des mots de passe pour l'authentification des utilisateurs finaux pour des serveurs distants.

Vérifiez aussi que les conditions suivantes sont remplies :

- Il existe au moins un ensemble d'abonnements pour le qualificatif Apply et l'ensemble d'abonnements contient un ou plusieurs des éléments suivants :
  - membre d'un ensemble d'abonnements,
  - instruction SQL,
  - Procédure.
- Toutes les tables cible condensées doivent posséder une clé cible, à savoir un ensemble de colonnes uniques (clé primaire ou index à entrées uniques) que le programme Apply emploie pour effectuer le suivi des changements qu'il réplique lors de chaque cycle du programme Apply. Les tables CCD non condensées ne possèdent pas de clés primaires ou d'index à entrées uniques.

#### A propos de cette tâche

Au démarrage du programme Apply, vous pouvez aussi indiquer des paramètres de démarrage.

#### Procédure

Pour démarrer le programme Apply :

Utilisez l'une des méthodes suivantes :

Option	Description
Centre de réplication	Utilisez la fenêtre Démarrage du programme Apply. Pour y accéder, ouvrez le dossier <b>Serveurs de contrôle d'Apply</b> dans la branche <b>Opérations</b> de l'arborescence des objets et cliquez sur le dossier <b>Qualificatifs Apply</b> . Dans le panneau du contenu, cliquez avec le bouton droit sur le qualificatif Apply représentant le programme Apply à démarrer et cliquez sur <b>Démarrage du programme Apply</b> .
<div style="background-color: #800000; color: white; padding: 2px; text-align: center;">z/OS</div> <div style="background-color: #800000; color: white; padding: 2px; text-align: center;">Linux UNIX Windows</div> Commande système <b>asnapply</b>	Utilisez cette commande pour démarrer le programme Apply.
<div style="background-color: #800000; color: white; padding: 2px; text-align: center;">z/OS</div> Console ou TSO z/OS	<p>Sous z/OS, vous pouvez démarrer le programme Apply avec JCL ou comme tâche lancée par le système. Vous pouvez indiquer de nouvelles valeurs de paramètres d'appel au démarrage d'un programme Apply avec JCL.</p> <p>La longueur totale des paramètres z/OS, que vous pouvez indiquer dans la zone PARMs=, 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 définis dans le fichier SYSIN sont automatiquement concaténés par le programme Apply aux paramètres PARMs=. Vous ne pouvez indiquer que des paramètres Apply dans le fichier SYSIN. Les paramètres LE doivent être indiqués dans la zone PARMs= ou LE _CEE_ENVFILE=DD, suivis d'une barre oblique (/).</p> <p><b>Exemple :</b></p> <pre> /** asterisk indicates a comment line // APP EXEC PGM=ASNAPP,PARMS='LE/Apply parameters' /** Parameters can be any or no LE parameters and any or /** no Apply parameters //SYSIN DD * /** additional Apply parameters, one or more /** parameters on each line     APPLY_SERVER=DSN!! APPLY_SCHEMA=APPCAT     DEBUG=Y LOGSTDOUT=N </pre>
<div style="background-color: #800000; color: white; padding: 2px; text-align: center;">Windows</div> Services Windows	Vous pouvez créer un service de réplication DB2 sur le système d'exploitation Windows afin de démarrer automatiquement le programme Q avec le système.

Une fois le programme Apply démarré, il s'exécute en continu (sauf si vous avez employé le paramètre de démarrage **copyonce**) jusqu'à ce que l'un des événements suivants se produise :

- Vous arrêtez le programme Apply à l'aide du Centre de réplication ou d'une commande.
- Le programme Apply ne peut pas se connecter au serveur de contrôle Apply.
- Le programme Apply ne peut pas allouer de mémoire pour le traitement.

Pour vérifier si un programme Apply est démarré, utilisez l'une des méthodes suivantes :

- **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.
- Recherchez dans le fichier journal de diagnostic Apply (*serveur\_apply.qual\_apply.APP.log* sur z/OS et *instancedb2.serveur\_apply.qual\_apply.APP.log* sur Linux, UNIX et Windows) un message indiquant que le programme capture des modifications.
- Recherchez dans la table IBMSNAP\_APPLYTRACE un message indiquant que le programme applique des modifications.
- Utilisez la fenêtre Messages Apply dans le Centre de réplication pour afficher un message indiquant que le programme a démarré. Pour ouvrir cette fenêtre, cliquez avec le bouton droit sur le qualificatif Apply dans le panneau du contenu identifiant le programme Apply dont vous voulez voir les messages, puis sélectionnez **Rapports > Messages Apply**.
- Utilisez la fenêtre Vérification de l'état dans le Centre de réplication ou la commande `asnacmd status` pour afficher l'état de toutes les unités d'exécution Apply. Pour ouvrir cette fenêtre, cliquez avec le bouton droit sur le qualificatif Apply dans le panneau du contenu identifiant le programme Apply à vérifier, puis sélectionnez **Vérification de l'état**.

---

## Démarrage d'un programme Apply (System i)

Vous pouvez démarrer une instance du programme Apply pour commencer à appliquer des données à vos cibles.

### Avant de commencer

Veillez à ce que votre système soit correctement configuré :

- Les connexions sont configurées sur tous les serveurs de réplication.
- Vous possédez l'autorisation appropriée.
- Les tables de contrôle sont créées.
- Les programmes de réplication sont configurés.

Vérifiez aussi que les conditions suivantes sont remplies :

- Il existe au moins un ensemble d'abonnements actif pour le qualificatif Apply et cet ensemble d'abonnements contient un ou plusieurs des éléments suivants :
  - membre d'un ensemble d'abonnements,
  - instruction SQL,
  - Procédure.
- Toutes les tables cible condensées doivent posséder une clé cible, à savoir un ensemble de colonnes uniques (clé primaire ou index à entrées uniques) que le programme Apply emploie pour effectuer le suivi des changements qu'il réplique lors de chaque cycle du programme Apply. Les tables CCD non condensées ne possèdent pas de clés primaires ou d'index à entrées uniques.

### Procédure

Pour démarrer un programme Apply, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Commande système STRDPAPY</b>	Utilisez la commande Start DPR Apply (STRDPAPY) pour démarrer un programme Apply sur votre système local.

Méthode	Description
Centre de réplication	Utilisez la fenêtre Démarrage du programme Apply. Pour y accéder, ouvrez le dossier <b>Serveurs de contrôle d'Apply</b> dans la branche <b>Opérations</b> de l'arborescence des objets et cliquez sur le dossier <b>Qualificatifs Apply</b> . Dans le panneau du contenu, cliquez avec le bouton droit sur le qualificatif Apply représentant le programme Apply à démarrer et cliquez sur <b>Démarrage du programme Apply</b> .

Lorsque vous avez démarré le programme Apply, il s'exécute de manière continue jusqu'à ce que l'une des conditions suivantes soit vraie :

- Vous avez démarré le programme à l'aide du paramètre de démarrage COPYONCE(\*YES).
- Vous avez indiqué ALWINACT(\*NO) et il n'y a pas de données à traiter.
- Vous arrêtez le programme Apply à l'aide du Centre de réplication ou d'une commande.
- Le programme Apply ne peut pas se connecter au serveur de contrôle Apply.
- Le programme Apply ne peut pas allouer de mémoire pour le traitement.

## Paramètres d'exploitation par défaut du programme Apply

Lorsque vous créez les tables de contrôle du programme Apply, les valeurs par défaut des paramètres d'exploitation Apply sont enregistrées dans la table IBMSNAP\_APPPARMS.

Les valeurs par défaut sont indiquées dans le tableau 9 et dans le tableau 10, à la page 145. z/OS Linux UNIX Windows

Tableau 9. Paramètres par défaut pour les paramètres d'exploitation Apply (z/OS, Linux, UNIX, Windows)

Paramètre d'exploitation	Valeur par défaut	Nom de colonne dans la table IBMSNAP_APPPARMS
apply_qual	Pas de valeur par défaut	APPLY_QUAL
apply_path	Répertoire dans lequel Apply a été démarré <sup>1</sup>	APPLY_PATH
caf	y <sup>5</sup>	non disponible
control_server	DB2DBDFT <sup>2</sup>	non disponible
copyonce	n <sup>3</sup>	COPYONCE
db2_subsystem	Pas de valeur par défaut <sup>4</sup>	non disponible
delay	6 secondes	DELAY
errwait	300 secondes	ERRWAIT
inamsg	y <sup>5</sup>	INAMSG
loadxit	n <sup>3</sup>	LOADXIT
logreuse	n <sup>3</sup>	LOGREUSE
logstdout	n <sup>3</sup>	LOGSTDOUT
notify	n <sup>3</sup>	NOTIFY
opt4one	n <sup>3</sup>	OPT4ONE
pwdfile	asnpwd.aut	non disponible

Tableau 9. Paramètres par défaut pour les paramètres d'exploitation Apply (z/OS, Linux, UNIX, Windows) (suite)

Paramètre d'exploitation	Valeur par défaut	Nom de colonne dans la table IBMSNAP_APPPARMS
spillfile	disque <sup>6</sup>	SPILLFILE
sleep	y <sup>5</sup>	SLEEP
sqlerrcontinue	n <sup>3</sup>	SQLERRCONTINUE
term	y <sup>5</sup>	TERM
trlreuse	n <sup>3</sup>	TRLREUSE

**Remarque :**

1. Si Apply démarre en tant que service Windows, son emplacement est sqllib\bin.
2. Le serveur de contrôle Apply correspond à la valeur de la variable d'environnement DB2DBDFT, si cette variable est spécifiée. Pour Linux, UNIX et Windows uniquement.
3. aucun
4. Le nom de sous-système DB2 entré peut contenir au maximum quatre caractères. Ce paramètre est obligatoire. Le nom du sous-système DB2 ne s'applique qu'aux systèmes d'exploitation z/OS.
5. oui
6. Sous z/OS, la valeur par défaut est MEM.

**System i**

Tableau 10. Configuration par défaut des paramètres d'exploitation Apply (System i)

Paramètre d'exploitation	Description (*valeur)
USER (*CURRENT)	Utilisateur connecté au système.
JOBID (*LIBL/QZSNDPR)	Nom de la bibliothèque produit / description de travail.
APYQUAL (*USER)	Nom de l'utilisateur actuel (ci-dessus).
CTLSVR (*LOCAL)	Nom de serveur RDB local.
TRACE (*NONE)	Ne pas générer de trace.
FULLREFPGM (*NONE)	Ne pas exécuter la routine d'exit ASNLOAD.
SUBNFYPGM (*NONE)	Ne pas exécuter la routine d'exit ASNDONE.
INACTMSG (*YES)	Lorsque le programme Apply commence une période inactive, il génère le message ASN1044, qui décrit la durée d'inactivité du programme.
ALWINACT (*YES)	Mise en veille si aucun élément n'est à traiter
DELAY (6)	Attente de 6 secondes après un cycle Apply pour reprendre un nouveau traitement.
RTYWAIT (300)	Attente de 300 secondes avant de faire une nouvelle tentative après un échec.
COPYONCE (*NO)	Ne pas arrêter avant de terminer un cycle de copie, poursuivre le traitement.
TRLREUSE (*NO)	Ne pas vider la table IBMSNAP_APPLYTRAIL lorsque le programme Apply démarre.
OPTSNGSET (*NO)	Ne pas optimiser les performances du programme Apply pour le traitement d'un seul ensemble d'abonnements.

---

## Description des paramètres d'exploitation Apply

Lorsque vous lancez le programme Apply, vous pouvez sélectionner des paramètres de démarrage. Voici les paramètres de démarrage ; pour chacun d'entre eux, des recommandations de choix de valeur sont mentionnées.

Ces paramètres s'appliquent à z/OS, Linux, UNIX, et Windows à moins que cela ne soit indiqué différemment.

- «apply\_path»
- «apply\_qual», à la page 147
- caf
- «control\_server», à la page 148
- «copyonce», à la page 148
- «Paramètre db2\_subsystem (z/OS)», à la page 149
- «delay», à la page 149
- «errwait», à la page 149
- «inamsg», à la page 150
- «loadxit», à la page 150
- «logreuse», à la page 150
- «logstdout», à la page 151
- «notify», à la page 151
- «opt4one», à la page 151
- «pwdfile», à la page 152
- «sleep», à la page 152
- «spillfile», à la page 153
- «sqlerrcontinue», à la page 153
- «term», à la page 154
- «trlreuse», à la page 154

### apply\_path

**Linux UNIX Windows** Valeur par défaut : `apply_path=répertoire_en_cours`

**Windows** Valeur par défaut (service sous Windows): `apply_path`  
`sqllib\bin`

Le chemin d'accès au programme Apply représente le répertoire dans lequel se trouvent le fichier journal et les fichiers de travail Apply. Par défaut, le chemin d'accès à Apply est le répertoire dans lequel vous lancez le programme. Vous pouvez modifier le chemin d'accès à Apply afin de conserver le fichier journal et les fichiers de travail à un autre emplacement (`/home/db2inst/apply_files`, par exemple sur un système AIX). Notez le répertoire choisi, car vous pourrez avoir besoin d'accéder à ce répertoire pour ouvrir le fichier journal.

Vous pouvez indiquer soit un nom de chemin, soit un qualificatif de haut niveau (HLQ), tel que `//APPV9`. Lorsque vous utilisez un qualificatif de haut niveau, des fichiers séquentiels sont créés conformément aux conventions de dénomination z/OS pour ce type de noms 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 à un emplacement explicitement nommé (le qualificatif HLQ

est dans ce cas concaténé en première partie du nom de fichier). Par exemple, sysadm.APPV9.nomfichier. L'utilisation d'un qualificatif HLQ peut être utile si vous voulez que les fichiers LOADMSG et les fichiers journaux Apply soit gérés par le système (SMS).

Si vous souhaitez que la tâche démarrée de Apply ait un droit en écriture sur un ensemble de données .log avec un ID utilisateur différent de l'ID à l'origine de l'exécution de la tâche (par exemple TSOUSER), vous devez indiquer une apostrophe (') comme caractère d'échappement lors de l'utilisation du format SYSIN pour les paramètres d'entrée dans la tâche démarrée. Par exemple, si vous voulez utiliser le qualificatif de haut niveau JOESMITH, l'ID utilisateur TSOUSER qui exécute le programme Apply doit alors disposer des droits RACF en écriture sur les ensembles de données avec le qualificatif de haut niveau JOESMITH, comme illustré :

```
//SYSIN DD *  
  APPLY_PATH=/'JOESMITH  
/*
```

**z/OS** Consultez le travail SASNSAMP(ASNSTRA) pour plus d'informations sur les méthodes de modification du chemin d'accès à Apply.

**Important :** Assurez-vous que le répertoire choisi contient une quantité suffisante d'espace pour les fichiers temporaires utilisés par le programme Apply.

**Windows** **Démarrage d'instances Apply sur un système Windows :** lorsque vous lancez le programme Apply en utilisant le Centre de réplication ou la commande **asnapply**, vous devez spécifier son chemin d'accès si deux qualificatifs Apply ou davantage sont identiques, sauf pour leur mise en majuscules. Les noms de fichiers sous Windows ne sont pas sensibles à la casse. Par exemple, supposons que vous possédiez trois qualificatifs Apply : APPLYQUAL1, ApplyQual1 et applyqual1. Chacune de ces instances Apply doit être démarrée à l'aide d'une valeur différente de paramètre **apply\_path**, afin d'empêcher les conflits de noms de fichiers journaux, pour chaque instance du programme Apply).

## apply\_qual

Vous devez spécifier le qualificatif Apply de tous les ensembles d'abonnements à traiter. (Vous avez défini ce qualificatif Apply lorsque vous avez créé les ensembles d'abonnements.) Vous ne pouvez spécifier qu'un qualificatif Apply par commande de démarrage.

**Important :** Le qualificatif Apply est sensible à la casse et la valeur entrée doit correspondre à la valeur de la colonne APPLY\_QUAL de la table IBMSNAP\_SUBS\_SET.

Si plusieurs qualificatifs Apply sont définis, vous pouvez lancer une autre instance du programme Apply. Chaque instance de programme Apply démarrée traite différents ensembles d'abonnements représentés sur le même serveur de contrôle Apply. Par exemple, supposons que deux ensembles d'abonnements soient définis et que chacun d'entre eux possède un qualificatif Apply unique : APPLY1 et APPLY2. Vous pouvez lancer deux instances de programme Apply (une pour chaque qualificatif Apply) ; chaque instance utilise les tables de contrôle situées sur le serveur de contrôle Apply (CNTRLSVR). Chaque instance Apply traite ses propres ensembles d'abonnement de façon indépendante, offrant ainsi de meilleures performances que si une seule instance Apply traitait tous les ensembles.

z/OS

## caf

Valeur par défaut : o

Le paramètre d'exécution **caf =y** indique si le programme Apply remplace la connexion RRS et s'exécute avec la connexion CAF. L'option **caf =y** est l'option par défaut du programme Apply.

## control\_server

z/OS

Valeur par défaut : aucune

Linux UNIX Windows

Valeur par défaut : La valeur de la variable d'environnement DB2DBDFT, si elle est disponible

Le serveur de contrôle Apply est le serveur sur lequel résident les les définitions d'abonnement et les tables de contrôle Apply. Un seul serveur de contrôle doit être spécifié par qualificatif Apply. Si vous n'indiquez aucune valeur, le programme Apply est démarré sur le serveur par défaut. La valeur par défaut dépend de votre système d'exploitation.

z/OS

Sous z/OS, vous devez indiquer le paramètre **serveur de contrôle**.

Si le programme Apply ne parvient pas à se connecter au serveur de contrôle, il suit l'action définie par le paramètre **term** :

### **term=y (valeur par défaut)**

Le programme Apply s'arrête.

### **term=n**

Le programme Apply attend le temps défini par le paramètre **errwait** et réessaie de se connecter.

L'unité d'exécution agent prend le statut "attente de la base de données" s'il ne peut pas se connecter à son serveur de contrôle Apply et que le programme Apply a été démarré avec le paramètre **term=n**. Vous pouvez exécuter la commande **asnacmd** ou **MODIFY** sous z/OS pour vérifier si l'unité d'exécution agent est en cours d'exécution mais qu'elle ne peut pas se connecter au serveur de contrôle.

Si le programme Apply ne parvient pas à se connecter aux autres serveurs, il génère un message d'erreur et poursuit son traitement.

## copyonce

Valeur par défaut : **copyonce=n**

Le paramètre **copyonce** détermine le cycle de copie du programme Apply.

Lorsque vous lancez le programme Apply en spécifiant **copyonce=y**, il traite chaque ensemble d'abonnements admissible une seule fois, puis il s'arrête. Dans ce cas, un ensemble d'abonnements peut être traité si l'une des conditions suivantes se vérifie :

- L'ensemble d'abonnements utilise l'heure relative, l'heure s'est écoulée et l'ensemble d'abonnements est activé.



- L'ensemble d'abonnements utilise l'heure basée sur l'événement, il est activé et l'événement s'est produit mais le programme Apply n'a pas encore traité l'ensemble d'abonnements.

En règle générale, le programme Apply est démarré à l'aide du paramètre **copyonce=n** car cela permet de poursuivre l'exécution de ce programme et le traitement des abonnements admissibles.

Si vous exécutez le programme Apply à partir d'un environnement connecté occasionnellement au réseau, utilisez **copyonce=y** au lieu de **copyonce=n**. Vous pouvez également utiliser **copyonce=y** si vous exécutez le programme Apply au sein d'un environnement de test.

**Conseil :** Utilisez **sleep=n** au lieu de **copyonce=y** si vous souhaitez que le programme Apply traite plusieurs fois chaque ensemble d'abonnements, sous réserve que cet ensemble soit admissible et que des données soient disponibles pour la réplication. La configuration **copyonce=y** permet de traiter chaque ensemble une seule fois, même s'il existe des données supplémentaires à répliquer.

## Paramètre **db2\_subsystem** (z/OS)

Le paramètre **db2\_subsystem** indique le nom du sous-système DB2, si vous exécutez Apply sous z/OS. Le nom de sous-système DB2 saisi doit contenir quatre caractères au maximum. Il n'existe pas de valeur par défaut pour ce paramètre. Ce paramètre est obligatoire.

### **delay**

**Valeur par défaut :** **delay=6** secondes

Le paramètre **delay** permet de définir une durée (exprimée en secondes) pendant laquelle le programme Apply attend la fin du cycle Apply.

Par défaut, au cours de la réplication continue (lorsque l'ensemble d'abonnements utilise **sleep=0** minutes), le programme Apply attend 6 secondes après le traitement d'un ensemble d'abonnements avant de faire une nouvelle tentative au niveau de cet ensemble. Utilisez une valeur non égale à zéro pour enregistrer les cycles d'UC lorsqu'il n'y a pas d'activité de base de données à répliquer. Utilisez une valeur de délai plus faible pour la réplication nécessitant peu de temps d'attente.

**Remarque :** Ce paramètre est ignoré si **copyonce** est indiqué.

### **errwait**

**Valeur par défaut :** **errwait=300** secondes (5 minutes)

Le paramètre **errwait** indique le nombre de secondes d'attente avant que le programme Apply ne tente de nouveau de traiter un ensemble d'abonnements après l'échec d'un cycle d'abonnement.

Par défaut, le programme Apply attend 300 secondes avant de faire une nouvelle tentative de traitement d'ensemble d'abonnements après l'échec d'un cycle d'abonnement. Vous pouvez utiliser une valeur plus faible en environnement de

test. La valeur minimale est égale à 1 seconde. Au sein d'un environnement de production, avant de modifier la valeur par défaut de ce paramètre, tentez des compromis :

- Si vous utilisez une valeur plus faible, vous risquez de gâcher des cycles d'UC si le programme Apply répète des erreurs permanentes. Par exemple, vous gâchez des cycles d'UC si le programme Apply tente sans cesse de traiter un ensemble d'abonnements alors qu'une table cible présente un problème. Un grand nombre de messages risque de s'afficher dans le fichier journal et, si le programme Apply est exécuté sous z/OS, sur la console de l'opérateur.
- Si vous utilisez une valeur plus élevée, vous risquez d'augmenter la latence si le programme Apply doit attendre pour faire une nouvelle tentative en raison d'erreurs transitoires. Par exemple, vous augmentez la latence si vous utilisez une valeur plus élevée pour le paramètre **errwait**, car le programme Apply attend inutilement après la survenue d'une erreur réseau qui peut être corrigée rapidement.

**Remarque :** Le paramètre **errwait** est ignoré si **copyonce** est indiqué.

## **inamsg**

**Valeur par défaut :** **inamsg=y**

Le paramètre **inamsg** indique si le programme Apply doit émettre un message lorsqu'il devient inactif.

Par défaut, le programme Apply émet un message lorsqu'il devient inactif. Si vous ne souhaitez pas que le programme Apply émette un message lorsqu'il devient inactif (car les messages occupent une grande quantité d'espace dans le fichier journal Apply), et tout particulièrement si le programme Apply n'attend pas longtemps entre les différents traitements d'ensembles d'abonnements. Pour désactiver ces messages, utilisez la commande **inamsg=n**.

## **loadxit**

**Valeur par défaut :** **loadxit=n**

Le paramètre **loadxit** indique si le programme Apply doit régénérer les tables cible à l'aide de la routine d'exit ASNLOAD.

Par défaut, le programme Apply n'utilise pas la routine d'exit ASNLOAD lorsqu'il effectue une régénération des tables cible (**loadxit=n**). Utilisez la commande **loadxit=y** si vous souhaitez que le programme Apply appelle la routine d'exit ASNLOAD pour la régénération des tables cible. Il est conseillé d'utiliser la routine d'exit ASNLOAD lorsqu'une grande quantité de données doit être copiée dans les tables cible au cours d'une régénération intégrale.

**z/OS** Sous z/OS, la routine d'exit ASNLOAD utilise la procédure stockée DSNUTILS pour appeler les fonctionnalités DB2 requises pour charger la table cible.

## **logreuse**

**Valeur par défaut :** **logreuse=n**

Le programme Apply conserve les informations opérationnelles dans un fichier journal. Ce paramètre indique s'il est préférable de l'ajouter au fichier journal ou de l'ignorer.

#### z/OS

Le nom du fichier journal est *serveur\_contrôle.qual\_apply.APP.log*

#### Linux UNIX Windows

Le nom du fichier journal est  
*instancedb2.serveur\_contrôle.qual\_apply.APP.log*.

Par défaut, le programme Apply ajoute des messages au fichier journal (**logreuse=n**) chaque fois que vous démarrez le programme Apply. Si vous souhaitez disposer de l'historique des messages émis par le programme Apply, conservez les valeurs par défaut. Dans les cas suivants, vous souhaitez peut-être utiliser **logreuse=y**, qui permet au programme Apply de supprimer le journal et de le créer de nouveau au moment du démarrage :

- La taille du journal devient volumineuse et vous souhaitez nettoyer le journal pour gagner de l'espace.
- Vous n'avez pas besoin de l'historique qui se trouve dans le journal.

## logstdout

Valeur par défaut : **logstdout=n**

Le paramètre **logstdout** n'est disponible que si vous utilisez la commande **asnapply** ; il n'est pas disponible dans le Centre de réplication.

Le paramètre **logstdout** indique si le programme Apply doit envoyer des messages d'opération terminée (ASN10251) au fichier journal et à la sortie standard.

Par défaut, le programme Apply n'envoie pas de messages d'opération terminée à la sortie standard (STDOUT). Si vous spécifiez **logstdout=y**, le programme Apply envoie des messages d'opération aboutie au fichier journal et à la sortie standard (STDOUT). Vous pouvez choisir d'envoyer ces messages à la sortie standard en cas d'identification d'incident ou si vous surveillez le fonctionnement du programme Apply.

## notify

Valeur par défaut : **notify=n**

Le paramètre **notify** indique si le programme Apply doit notifier la routine d'exit ASNDONE après le traitement d'un abonnement.

Par défaut, le programme Apply ne notifie pas la routine d'exit ASNDONE lorsque le traitement des abonnements est terminé. Si vous spécifiez **notify=y**, une fois que le programme Apply a terminé un cycle d'abonnement, il appelle ASNDONE pour effectuer des traitements supplémentaires (tels que l'examen des tables de contrôle Apply ou l'envoi de courriers électroniques).

## opt4one

Valeur par défaut : **opt4one=n**

Le paramètre **opt4one** indique si le programme Apply est optimisé pour un ensemble d'abonnements.

**Remarque :** Le paramètre **opt4one** est ignoré si **copyonce** est indiqué.

Par défaut, le programme Apply est optimisé pour un grand nombre d'ensembles d'abonnements. Le programme Apply lit les informations contenues dans les tables de contrôle au début de chacun des cycles de copie. Si un ensemble d'abonnements est défini pour le qualificatif Apply, démarrez le programme Apply à l'aide de **opt4one=y**, afin que le programme Apply place en mémoire cache les informations sur les membres d'ensembles d'abonnement et les colonnes, et qu'il les réutilise. Lorsque vous optimisez le programme Apply pour un ensemble d'abonnements, le programme Apply utilise une quantité moins élevée d'UC et le rendement augmente.

**Important :** lorsque vous utilisez **opt4one=y** et que vous ajoutez un membre à un ensemble ou que vous modifiez un ensemble, vous devez arrêter le programme Apply et le redémarrer, afin qu'il puisse extraire les données dans les tables de contrôle.

## **pwdfile**

**Valeur par défaut :** **pwdfile=asnpwd.aut**

Si vos données sont réparties sur plusieurs serveurs, vous pouvez stocker les ID utilisateur et les mots de passe dans un fichier de mots de passe codés, afin que le programme Apply puisse accéder aux données situées sur des serveurs distants.

## **sleep**

**Valeur par défaut :** **sleep=y**

Le paramètre **sleep** indique si le programme Apply doit poursuivre son fonctionnement en mode d'économie d'énergie ou s'arrêter une fois le traitement des ensembles d'abonnements terminé.

Par défaut, le programme Apply démarre avec **sleep=y**. Il vérifie les ensembles d'abonnements admissibles. S'il trouve un ensemble d'abonnements admissible, il le traite et poursuit sa recherche d'autres ensembles admissibles. Le programme Apply continue de traiter les ensembles d'abonnements admissibles au fur et à mesure qu'il les trouve. Lorsqu'il ne parvient plus à trouver d'ensembles d'abonnements admissibles, il fonctionne en mode d'économie d'énergie et se 'réveille' périodiquement pour rechercher des ensembles d'abonnements admissibles. En règle générale, le programme Apply est démarré de cette façon, car les mises à jour sont ainsi appliquées au fur et à mesure et le programme Apply reste actif.

**Remarque :** Le paramètre **sleep** est ignoré si **copyonce** est indiqué.

Lorsque vous démarrez le programme Apply avec **sleep=n**, il recherche les ensembles d'abonnements admissibles et les traite. Il continue de traiter les ensembles d'abonnements admissibles jusqu'à ce qu'il ne puisse plus en trouver, et répète cette opération jusqu'à ce qu'il n'y ait plus aucune donnée à répliquer ; ensuite, il s'arrête. En règle générale, vous devez utiliser **sleep=n** au sein d'un environnement mobile ou d'un environnement de test, dans lequel le programme Apply fonctionne uniquement s'il trouve des ensembles d'abonnements

admissibles, puis s'arrête. Vous ne souhaitez généralement pas que le programme Apply reste en mode d'économie d'énergie et se réveille périodiquement pour rechercher d'autres ensembles admissibles. Dans ces environnements, vous souhaitez contrôler le moment où le programme Apply est exécuté, plutôt que de le laisser fonctionner indéfiniment.

**Conseil :** Utilisez `copyonce=y` au lieu de `sleep=n` si vous souhaitez traiter une seule fois chaque ensemble d'abonnements.

## spillfile

z/OS

Valeur par défaut : `spillfile=MEM`

Linux UNIX Windows

Valeur par défaut : `spillfile=disk`

Le programme Apply extrait les données des tables source et les place dans un fichier auxiliaire situé sur le système sur lequel le programme Apply fonctionne.

z/OS

Sous z/OS, le fichier auxiliaire est stocké par défaut dans la mémoire. Si vous demandez à ce que le fichier auxiliaire soit stocké sur disque, le programme Apply utilise les spécifications de l'instruction DD ASNASPL DD pour allouer les fichiers auxiliaires. Si l'instruction DD ASNASPL DD n'est pas spécifiée, il utilise VIO. Vous pouvez également spécifier `spillfile=hs` et le programme Apply utilisera l'espace de données hautes performances (hiperspace) pour déverser les données.

Linux UNIX Windows

Les seules valeurs valides pour `spillfile` sont `disk` (car les fichiers auxiliaires sont toujours situés sur disque, à l'emplacement spécifié par le paramètre `apply_path`).

## sqlerrcontinue

Valeur par défaut : `sqlerrcontinue=n`

Le paramètre `sqlerrcontinue` indique si le programme Apply doit réagir à certaines erreurs.

Par défaut, lorsque le programme Apply détecte une erreur SQL, il arrête le traitement de l'ensemble d'abonnements et génère un message d'erreur. En règle générale, vous devez utiliser la valeur par défaut dans votre environnement de production.

Si vous vous trouvez en environnement de test, vous pouvez vous attendre à ce que certaines erreurs SQL se produisent lors de l'insertion des données dans les tables cible. Ces erreurs peuvent être acceptables, mais elles entraînent l'arrêt du cycle d'abonnement en cours. Dans ces situations, vous pouvez démarrer le programme Apply avec `sqlerrcontinue=y` afin que celui-ci ignore ces erreurs et n'annule pas les données répliquées de ce cycle. Si le programme Apply reçoit un message d'erreur SQL lors de l'insertion des données dans une table cible, il vérifie les valeurs du fichier `qual_apply.sqs`. S'il trouve une correspondance, il enregistre les détails sur l'erreur dans un fichier de messages d'erreur (`qual_apply.err`) et poursuit le traitement. Si le programme Apply détecte une erreur SQL ne figurant pas dans le fichier `qual_apply.sqs`, il arrête le traitement de l'ensemble et accède à l'ensemble suivant.

Avant de démarrer le programme Apply avec l'option **sqlerrcontinue=y**, vous devez créer le fichier *qual\_apply.sqs* et l'enregistrer dans le répertoire dans lequel vous appelez le programme Apply. Indiquez 20 valeurs au maximum, l'une après l'autre, dans le fichier. Si vous modifiez le contenu de ce fichier lorsque le programme Apply fonctionne, arrêtez Apply et redémarrez-le afin qu'il reconnaisse les nouvelles valeurs.

**Exemple** : supposons que vous souhaitiez que le programme Apply poursuive le traitement d'un ensemble d'abonnements si une table cible reçoit le message d'erreur suivant (sqlstate/code) :

**23505/-803**

Violation de duplication d'index

Vous devez créer un fichier des états SQL contenant l'état SQL suivant :

23505

Si l'état SQL est retourné lors de la mise à jour de la table cible, le programme Apply applique les modifications aux autres tables cible de l'ensemble et crée un fichier de messages d'erreur qui décrit l'erreur elle-même, ainsi que les lignes rejetées.

**Conseil** : Vérifiez la colonne STATUS de la table IBMSNAP\_APPLYTRAIL. La valeur 16 indique que le programme Apply a traité l'ensemble d'abonnements avec succès, mais que certaines erreurs autorisées (définies dans le fichier *qual\_apply.sqs* se sont produites.

## **term**

**Valeur par défaut** : **term=y**

Le paramètre **term** détermine les actions du programme Apply s'il ne peut pas se connecter à son serveur de contrôle.

Par défaut, le programme Apply s'arrête s'il ne peut pas se connecter.

Utilisez le paramètre **term=n** si vous voulez que le programme continue de fonctionner. Celui-ci consigne une erreur, attend le temps défini par le paramètre **errwait** et réessaie de se connecter.

Le paramètre **term** est ignoré si **copyonce** est indiqué.

## **trlreuse**

**Valeur par défaut** : **trlreuse=n**

Le paramètre **trlreuse** indique si la table IBMSNAP\_APPLYTRAIL doit être réutilisée (ajoutée) ou écrasée lorsque le programme Apply démarre.

Par défaut, lorsque le programme Apply démarre, il ajoute des entrées à la table récapitulative Apply. Cette table contient l'historique des opérations effectuées sur toutes les instances Apply sur le serveur de contrôle Apply. Il s'agit d'un référentiel de diagnostic et de statistiques sur les performances. Conservez les valeurs par défaut si vous souhaitez disposer de l'historique des mises à jour. Dans les cas suivants, vous souhaitez peut-être que le programme Apply vide la table récapitulative Apply lorsqu'il démarre, au lieu de l'ajouter (**trlreuse=y**) :

- La table récapitulative Apply devient trop volumineuse et vous souhaitez la nettoyer pour gagner de l'espace.
- Vous n'avez pas besoin de l'historique qui se trouve dans la table.

**Conseil :** Au lieu d'utiliser `tr1reuse=y`, vous pouvez utiliser SQL une fois que le programme Apply a terminé le traitement d'un ensemble d'abonnements (`status=0`) pour supprimer des lignes de la table récapitulative Apply.

---

## Méthodes de modification des paramètres d'exploitation Apply

Vous pouvez modifier les valeurs par défaut des paramètres d'exploitation en spécifiant des valeurs généralement utilisées dans votre environnement. Vous pouvez également remplacer ces valeurs par défaut au moment du démarrage du programme Apply.

### Définition de nouvelles valeurs par défaut dans la table IBMSNAP\_APPPARMS

La table IBMSNAP\_APPPARMS contient des paramètres que vous pouvez modifier pour contrôler le fonctionnement du programme Apply. Une fois la table créée, elle contient les valeurs par défaut applicables au programme Apply.

### Indication des valeurs de paramètres au moment du démarrage du programme

**Apply** Vous pouvez indiquer des valeurs applicables à Apply au moment du démarrage de ce programme. Les valeurs que vous définissez au cours du démarrage contrôlent le comportement d'Apply pour la session en cours et remplacent les valeurs des paramètres d'exploitation par défaut, ainsi que les valeurs présentes dans la table de paramètres Apply. Elles ne mettent pas à jour les valeurs de la table de paramètres Apply. Si vous ne modifiez pas la table de paramètres Apply avant de démarrer le programme Apply et que vous ne spécifiez aucun paramètre d'exploitation lors du démarrage du programme Apply, des valeurs par défaut sont utilisées pour les paramètres d'exploitation.

Linux UNIX Windows

### Exemple

Supposons que vous ne souhaitiez pas utiliser les paramètres par défaut pour **errwait** pour le qualificatif Apply ASNPROD. Mettez à jour la table de paramètres au niveau du qualificatif Apply ASNPROD. Affectez à l'intervalle **errwait** la valeur 600 secondes.

```
update asn.ibmsnap_appparms set errwait=600 where apply_qual='ASNPROD'
```

---

## Modification des paramètres Apply enregistrés dans la table IBMSNAP\_APPPARMS (Linux, UNIX, Windows, z/OS)

La table IBMSNAP\_APPPARMS contient les paramètres d'exploitation enregistrés destinés au programme Apply. Lorsque vous démarrez le programme Apply, ce dernier utilise les valeurs de cette table, sauf si vous les remplacez temporairement en utilisant des paramètres de démarrage.

### A propos de cette tâche

Une seule ligne est autorisée pour chaque qualificatif Apply. Si vous souhaitez modifier une ou plusieurs valeurs par défaut, vous pouvez mettre à jour les

colonnes au lieu d'insérer des lignes. Si vous supprimez une ligne, le programme Apply utilise les valeurs par défaut, sauf si ces valeurs sont remplacées par les valeurs des paramètres de démarrage.

Le programme Apply ne lit cette table qu'au cours du démarrage : par conséquent, vous devez arrêter puis redémarrer le programme Apply si vous souhaitez l'exécuter avec les nouveaux paramètres. Si vous modifiez la table de paramètres Apply pendant que ce programme est en cours d'exécution, cela ne modifie pas le fonctionnement du programme.

---

## Arrêt du programme Apply

Lorsque vous arrêtez le programme Apply, il ne copie plus de données dans les tables cible et il met à jour les tables de contrôle pour garantir un prochain démarrage propre.

### Procédure

Pour arrêter le programme Apply :

Utilisez l'une des méthodes suivantes :

Option	Description
<b>Centre de répliation</b>	Utilisez la fenêtre Arrêt du programme Apply. Pour y accéder, ouvrez le dossier <b>Serveurs de contrôle d'Apply</b> dans la branche <b>Opérations</b> de l'arborescence des objets et cliquez sur le dossier <b>Qualificatifs Apply</b> . Dans le panneau du contenu, cliquez avec le bouton droit sur le qualificatif Apply représentant le programme Apply à arrêter et cliquez sur <b>Arrêt du programme Apply</b> .
<b>z/OS</b> <b>Linux UNIX Windows</b> Commande système <b>asnacmd stop</b>	Utilisez cette commande pour arrêter le programme Apply.
<b>System i</b> Commande système <b>ENDDPRAPY</b>	Utilisez la commande End DPR Apply (ENDDPRAPY) pour arrêter un programme Apply sur votre système local.

---

## Emplacement de stockage des détails relatifs aux échecs d'intégrité référentielle par le programme Apply

Vous pouvez consulter les détails relatifs aux lignes qui n'ont pas été appliquées aux tables cible en raison de violations de l'intégrité référentielle. Dans la répliation SQL, le programme Apply écrit ces détails dans un fichier.

Lorsque vous démarrez le programme Apply, il crée un fichier dans le chemin spécifié par le paramètre **apply\_path**, avec le nom suivant : *apply\_qualifier*.RI. Si le fichier existe déjà lors du démarrage du programme Apply, il efface le fichier et commence à écrire dans le fichier vide si nécessaire.



Si une opération INSERT, UPDATE ou DELETE sur une table cible échoue avec SQLCODE SQL0530, SQL0531 ou SQL0532 en raison d'une contrainte d'intégrité référentielle, le programme Apply écrit les informations relatives à la ligne dans le fichier.

Apply écrit également les informations suivantes dans le fichier, pour chaque ligne associée à une erreur :

- Nom d'ensemble d'abonnements
- SQLCA et SQLDA pour l'instruction erronée
- Instruction SQL ayant échoué
- Valeurs des colonnes IBMSNAP\_COMMITSEQ et IBMSNAP\_INTENTSEQ de la ligne de table de modification des données à l'origine de l'erreur

Lors de l'échec d'une opération sur la ligne, au niveau de la table cible, le programme Apply écrit la ligne défectueuse dans un fichier de relance distinct, puis il tente à nouveau l'opération. Une violation d'intégrité référentielle peut parfois être résolue de cette manière, ce qui vous évite d'intervenir. Si des erreurs d'intégrité référentielle continuent à se produire sur toutes les lignes du fichier de relance du programme Apply, ce dernier émet le message ASN0999E et le cycle Apply en cours échoue. Dans ce cas, vous devez éventuellement vérifier le fichier d'intégrité référentielle pour identifier la cause des erreurs.

---

## Modification de la routine d'exit ASNDONE (Linux, UNIX, Windows, z/OS)

Vous pouvez personnaliser la routine d'exit ASNDONE sous Linux, UNIX, Windows et z/OS afin de modifier le comportement du programme Apply après le traitement des abonnements.

### A propos de cette tâche

Si vous démarrez le programme Apply avec le paramétrage **notify=y**, Apply appelle la routine d'exit ASNDONE une fois le traitement des abonnements terminé (que ce traitement ait abouti ou échoué). La liste suivante fournit certains exemples de modification de la routine d'exit ASNDONE à utiliser dans votre environnement de réplication :

- Utilisez la routine d'exit afin d'examiner la table UOW pour les transactions rejetées et exécutez les actions requises (par exemple, envoi automatique d'un courrier électronique à l'opérateur de réplication, émission d'un message ou création d'une alerte) en cas de détection d'une transaction rejetée.
- Utilisez la routine d'exit pour désactiver un ensemble d'abonnements ayant échoué, afin que le programme Apply évite de tenter une nouvelle fois de traiter cet ensemble avant que le problème ne soit résolu. Pour détecter un ensemble d'abonnements ayant échoué, modifiez la routine d'exit pour qu'elle examine l'élément STATUS=-1 de la table IBMSNAP\_APPLYTRAIL. Pour désactiver l'ensemble d'abonnements, configurez la routine d'exit de telle sorte qu'elle spécifie ACTIVATE=0 dans la table IBMSNAP\_SUBS\_SET.
- Utilisez la routine d'exit pour manipuler les données une fois qu'elles ont été appliquées pour chaque ensemble d'abonnements. (Vous avez également la possibilité de définir des instructions d'exécution, à l'aide d'instructions SQL ou de procédures stockées, exécutées avant ou après que le programme Apply ait traité un ensemble d'abonnements spécifique.)

### Procédure

Pour utiliser une version modifiée de l'exemple de routine d'exit ASNDONE, procédez comme suit :

1. Modifiez la routine ASNDONE afin qu'elle réponde à vos besoins.
  - **z/OS** Consultez la section PROLOG de l'exemple de programme SASNSAMP(ASNDONE).
  - **Linux UNIX Windows** Pour plus d'informations sur la méthode de modification de cette routine d'exit, consultez la section PROLOG de l'exemple de programme (\sql11ib\samples\repl\asndone.smp).
2. Compilez et effectuez la liaison du programme, puis insérez le fichier exécutable dans le répertoire approprié.
3. Démarrez le programme Apply avec le paramétrage **notify=y** pour appeler la routine d'exit ASNDONE.

---

## Modification de la routine d'exit ASNDONE (System i)

Vous pouvez personnaliser la routine d'exit ASNDONE sur les systèmes d'exploitation System i afin de modifier le comportement du programme Apply une fois qu'il a terminé le traitement des abonnements.

### A propos de cette tâche

Si vous démarrez le programme Apply avec le paramètre SUBNFYPGM défini sur le nom de la routine d'exit ASNDONE, le programme Apply appelle la routine d'exit ASNDONE après avoir terminé le traitement des abonnements, que les abonnements aient ou non été traités avec succès. La liste suivante fournit certains exemples de modification de la routine d'exit ASNDONE à utiliser dans votre environnement de réplication :

- Utilisez la routine d'exit afin d'examiner la table UOW pour les transactions rejetées et exécutez les actions requises (par exemple, envoi automatique d'un courrier électronique à l'opérateur de réplication, émission d'un message ou création d'une alerte) en cas de détection d'une transaction rejetée.
- Utilisez la routine d'exit pour désactiver un ensemble d'abonnements ayant échoué, afin que le programme Apply évite de tenter une nouvelle fois de traiter cet ensemble avant que le problème ne soit résolu. Pour détecter un ensemble d'abonnements ayant échoué, modifiez la routine d'exit pour qu'elle examine l'élément STATUS= -1 de la table IBMSNAP\_APPLYTRAIL. Pour désactiver l'ensemble d'abonnements, configurez la routine d'exit de telle sorte qu'elle spécifie ACTIVATE=0 dans la table IBMSNAP\_SUBS\_SET.
- Utilisez la routine d'exit pour manipuler les données une fois qu'elles ont été appliquées pour chaque ensemble d'abonnements. (Vous avez également la possibilité de définir des instructions de traitement à l'aide d'instructions SQL ou de procédures stockées exécutées avant ou après que le programme Apply a traité un ensemble d'abonnements spécifique.)

### Procédure

Pour utiliser une version modifiée de l'exemple de routine d'exit ASNDONE, procédez comme suit :

1. Modifiez la routine ASNDONE afin qu'elle réponde à vos besoins. Le tableau 11 , à la page 159 indique où vous pouvez trouver le code source de cette routine dans les langages C, COBOL, et RPG :

Tableau 11. Code source pour ASNDONE

Langage du compilateur	Nom de bibliothèque	Nom de fichier source	Nom de membre
C	QDP4	QCSRC	ASNDONE
COBOL	QDP4	QCBLESRC	ASNDONE
RPG	QDP4	QRPGLESRC	ASNDONE

Lors de la modification du programme, tenez compte de ces aspects du groupe d'activation :

**Si le programme est créé afin de s'exécuter avec un nouveau groupe d'activation**

Le programme Apply et le programme ASNLOAD ne partageront pas les ressources SQL, telles que les connexions de base de données relationnelle et les ouvertures de fichier. Le code de gestion de l'activation dans le système d'exploitation System i libère toutes ressources allouées par le programme ASNLOAD avant que le contrôle ne soit renvoyé au programme Apply. Une ressource supplémentaire est utilisée chaque fois que le programme Apply appelle le programme ASNLOAD.

**Si le programme est créé afin de s'exécuter dans le groupe d'activation de l'appelant**

Il partage les ressources SQL avec le programme Apply. Concevez le programme de manière à minimiser son impact sur le programme Apply. Par exemple, le programme pourrait provoquer un traitement imprévu du programme Apply s'il modifie la connexion actuelle de la base de données relationnelle.

**Si le programme est créé pour s'exécuter dans un groupe d'activation nommé**

Il ne partage pas les ressources avec le programme Apply. Utilisez un groupe d'activation nommé pour éviter le temps système du groupe d'activation chaque fois que le programme ASNLOAD est appelé. Les structures de données d'exécution ainsi que les ressources SQL peuvent être partagées entre les appels. Le traitement de nettoyage de l'application n'est pas effectué tant que le programme Apply n'est pas terminé. Vous devez donc concevoir le programme de notification d'abonnement de manière à s'assurer qu'il ne provoque pas de conflit d'accès avec le programme Apply en laissant les tables source, les tables cible ou les tables de contrôle verrouillées lorsque le contrôle est renvoyé au programme Apply.

2. Compilez et effectuez la liaison du programme, puis insérez le fichier exécutable dans le répertoire approprié.
3. Démarrez le programme Apply et spécifiez le nom du programme ASNDONE en utilisant le paramètre SUBNFYPGM de la commande STRDPRAPY.

Par exemple, si le programme est nommé ASNDONE\_1 et réside dans la bibliothèque APPLIB, utilisez la commande suivante :

```
SUBNFYPGM(APPLIB/ASNDONE_1)
```

---

## Actualisation des tables cible à l'aide de la routine d'exit ASNLOAD

Vous pouvez utiliser la routine d'exit ASNLOAD pour effectuer une actualisation complète des tables cible ; cette méthode est plus efficace que la méthode classique d'Apply, qui consiste à charger les données dans les tables cible.

Par défaut, le programme Apply n'utilise pas la routine d'exit ASNLOAD lorsqu'il effectue une actualisation complète de chaque table cible d'un ensemble d'abonnements. Il procède à une sélection complète de la table source, puis transfère les données dans un fichier auxiliaire situé sur le serveur sur lequel fonctionne le programme Apply ; ensuite, il utilise des instructions INSERT pour remplir la table cible. Si vos tables source sont volumineuses, il est conseillé d'utiliser la routine d'exit ASNLOAD.

L'exemple de routine d'exit diffère pour chaque plateforme DB2 afin de tenir compte des options propres à chaque plateforme:

**z/OS**

**Linux UNIX Windows**

La routine d'exit ASNLOAD est fournie en tant qu'exemple de routine d'exit dans un format source et dans un format compilé.

**System i**

ASNLOAD est expédié uniquement en format source.

Si une erreur se produit lorsque le programme Apply appelle la routine d'exit ASNLOAD, le programme Apply émet un message, arrête le traitement de l'ensemble d'abonnements en cours, puis traite l'ensemble d'abonnements suivant.

## Actualisation des tables cible à l'aide de la routine d'exit ASNLOAD (Linux, UNIX, Windows)

Vous pouvez utiliser la routine d'exit ASNLOAD pour actualiser les tables cible plus efficacement sous Linux, UNIX et Windows. Vous pouvez également modifier la routine avant de l'utiliser.

### Avant de commencer

- Les tables cible doivent contenir uniquement des colonnes faisant partie du mappage de réplication.
- L'ID utilisateur entré pour le programme Apply doit être celui de l'instance DB2 sur laquelle la routine ASNLOAD est exécutée. Par exemple, sous Linux et UNIX, assurez-vous que l'instance DB2 et l'ID utilisateur Apply sont membres d'un groupe commun. Ensuite, définissez les droits afin que le répertoire de démarrage Apply puisse fournir un accès en écriture à l'instance DB2 à l'aide de la commande **chmod 775**.

### Restrictions

La routine d'exit ASNLOAD fonctionne avec les utilitaires EXPORT, IMPORT et LOAD (y compris la fonction LOAD FROM CURSOR). La fonction LOAD FROM CURSOR constitue l'option par défaut utilisée par la routine d'exit ASNLOAD si la source d'un membre d'ensemble d'abonnements est un pseudonyme, ou encore si la base de données cible est la même que la base de données source. La fonction LOAD FROM CURSOR peut également être utilisée avec les sources de données DB2 si les opérations suivantes ont été effectuées :

- Un pseudonyme de table source a été créé dans la base de données cible.
- Des colonnes de la table IBMSNAP\_SUBS\_MEMBR pour le membre d'ensemble d'abonnements ont été définies pour spécifier l'utilisation de la fonction LOAD FROM CURSOR. La valeur de ces colonnes peut être définie via le Centre de réplication :
  - La colonne LOADX\_TYPE doit indiquer que la fonction LOAD FROM CURSOR est à utiliser.

- Les colonnes LOADX\_SRC\_N\_OWNER et LOADX\_SRC\_N\_TABLE doivent indiquer les informations de pseudonyme source relatives au membre d'ensemble d'abonnements incluant la table source.

### A propos de cette tâche

Lorsque vous appelez l'exemple de routine d'exit, celui-ci choisit par défaut l'utilitaire à utiliser en fonction du serveur source, du serveur cible et de l'environnement d'exécution. La routine peut utiliser l'utilitaire EXPORT de DB2 avec l'utilitaire IMPORT de DB2 ou avec l'utilitaire LOAD de DB2 ; elle peut également se servir de l'utilitaire LOAD FROM CURSOR.

Vous pouvez utiliser la routine d'exit compilée, configurer son comportement via la personnalisation de la configuration de réplication, ou encore personnaliser le code d'exit lui-même. Vous pouvez personnaliser la configuration de réplication en mettant à jour les colonnes de la table IBMSNAP\_SUBS\_MEMBR ou un exemple de fichier de configuration (asnload.ini).

Pour utiliser la routine ASNLOAD comme indiqué, démarrez le programme Apply en utilisant le paramètre **loadxit=y**.

### Procédure

Pour utiliser une version modifiée de la routine d'exit ASNLOAD, procédez comme suit :

1. Modifiez la routine ASNLOAD pour qu'elle réponde aux besoins de votre site. Pour plus d'informations sur la méthode de modification de cette routine d'exit, consultez la section PROLOG de l'exemple de programme (\sqlib\samples\repl\asnload.smp).  
**Important** : l'exemple de source utilise des combinaisons d'ID utilisateur et de mot de passe issues du fichier asnload.ini. Si le fichier asnload.ini ne contient pas d'ID utilisateur et de mot de passe pour un serveur, ou si ce fichier n'est pas disponible, la routine d'exit tente de se connecter sans le paramètre **user** ou **using**.
2. Compilez et effectuez la liaison du programme, puis insérez le fichier exécutable dans le répertoire approprié.
3. Affectez la valeur 2 à LOADX\_TYPE pour les membres qui seront renseignés à l'aide du code que vous avez fourni.
4. Lancez le programme Apply avec le paramètre loadxit=y pour appeler la routine d'exit ASNLOAD.

La routine d'exit ASNLOAD génère les fichiers suivants dans le répertoire **apply\_path** de l'instance Apply qui a appelé la routine d'exit ASNLOAD :

#### **asnload** *qual\_apply.trc*

Ce fichier contient les informations de trace si la fonction de trace est activée. La routine d'exit ASNLOAD crée ce fichier. Si le fichier existe, les informations sont ajoutées à ce dernier.

#### **asnload** *qual\_apply.msg*

Ce fichier contient les messages généraux sur les échecs d'exit, les messages d'avertissement et les messages d'informations, y compris les statistiques de chargement. La routine d'exit ASNLOAD crée ce fichier. Si le fichier existe, les informations sont ajoutées à ce dernier.

**asnaEXPT *qual\_apply.msg***

Ce fichier contient les messages d'erreur, les messages d'avertissement ou les messages d'informations émis par l'utilitaire EXPORT de DB2. La routine d'exit ASNLOAD crée ce fichier. Si le fichier existe, les informations sont ajoutées à ce dernier.

**asnaIMPT *qual\_apply.msg***

Ce fichier contient les messages d'erreur, les messages d'avertissement ou les messages d'informations émis par l'utilitaire IMPORT de DB2. La routine d'exit ASNLOAD crée ce fichier. Si le fichier existe, les informations sont ajoutées à ce dernier.

**asnaLOAD *qual\_apply.msg***

Ce fichier contient les messages d'erreur, les messages d'avertissement ou les messages d'informations émis par l'utilitaire LOAD de DB2. La routine d'exit ASNLOAD crée ce fichier. Si le fichier existe, les informations sont ajoutées à ce dernier.

## Actualisation des tables cible à l'aide de la routine d'exit ASNLOAD (z/OS)

Vous pouvez utiliser la routine d'exit ASNLOAD pour actualiser les tables cible de manière plus efficace sur les systèmes d'exploitation z/OS. Vous pouvez également modifier la routine avant de l'utiliser.

### Avant de commencer

Les tables cible doivent contenir uniquement des colonnes faisant partie du mappage de réplication.

### A propos de cette tâche

La routine d'exit ASNLOAD appelle la fonctionnalité LOAD FROM CURSOR qui est disponible avec la Suite d'utilitaires DB2 V7 (ou version supérieure). La fonctionnalité effectue des extractions basées sur curseur pour obtenir des données de la source et charge ces données dans la cible.

La routine d'exit utilise LOAD avec LOG NO et réinitialise le statut COPYPEND de l'espace table. Vous pouvez modifier l'échantillon de code source ASNLOAD pour modifier les options de chargement. La source se compose de deux fichiers d'en-tête et de trois programmes C++.

Pour utiliser la routine ASNLOAD fournie, lancez le programme Apply à l'aide du paramètre **loadxit=y**.

### Procédure

Pour utiliser une version modifiée de la routine d'exit ASNLOAD, procédez comme suit :

1. Modifiez la routine afin qu'elle soit conforme aux exigences de votre site. Pour plus d'informations sur la méthode de modification de cette routine d'exit, consultez la section PROLOG de l'exemple de programme SASNAMP (ANSLOAD).
2. Compilez et effectuez la liaison du programme, puis insérez le fichier exécutable dans le répertoire approprié.
  - a. Assurez-vous que les conditions suivantes sont satisfaites :

- DB2 Universal Database for z/OS and OS/390 version 7 ou ultérieure, avec un support d'utilitaire, est installé.
  - La procédure stockée DSNUTILS est en cours d'exécution. DSNUTILS doit s'exécuter dans un environnement WLM. Pour plus d'informations sur l'utilisation de DSNUTILS, voir *DB2 for z/OS V8 Utility Guide and Reference*.
- b. Utilisez l'exemple de fichier `zmac` (`SASNSAMP(ASNCMPLD)`) pour compiler et effectuer une édition de lien du programme d'exit utilisateur `ASNLOAD` dans `USS`.
  - c. Associez la routine d'exit `ASNLOAD` avec `DSNUTILS` et le module `Apply`. L'exemple d'`ASNLOAD` exécute la charge avec `LOG NO` puis répare l'espace table pour définir `nocopypend`. Il ne sauvegarde pas les espaces table. Par défaut, `ASNLOAD` crée deux fichiers temporaires sous l'ID utilisateur qui exécute l'instance du programme `Apply`, sauf si le paramètre **`apply_path`** avec l'option `APPLY_PATH=//` est spécifié pour cette instance d'`Apply`. Si tel est le cas, deux fichiers temporaires seront créés sous le qualificatif de haut niveau indiqué dans `APPLY_PATH`. La routine crée également un fichier contenant toutes les informations concernant la charge.
3. Affectez la valeur `loadx_type = 2` pour les membres qui seront renseignés à l'aide du code que vous avez fourni.
  4. Démarrez le programme `Apply` avec le paramètre **`loadxit=y`** pour appeler la routine d'exit `ASNLOAD`.

La routine d'exit `ASNLOAD` génère les fichiers suivants dans le répertoire **`apply_path`** ou `HLQ` pour l'instance d'`Apply` qui a appelé la routine d'exit `ASNLOAD` :

*IDutilisateur.qual\_apply.LOADMSG*

Ce fichier contient des messages d'incident, d'avertissement et d'information dont notamment les statistiques de charge. La routine d'exit `ASNLOAD` crée ce fichier. Si le fichier existe, les informations sont ajoutées à ce dernier.

*IDutilisateur.qual\_apply.LOADTRC*

Ce fichier contient les informations de trace si la fonction de trace est activée. La routine d'exit `ASNLOAD` crée ce fichier. Si le fichier existe, les informations sont ajoutées à ce dernier.

## Personnalisation du comportement de l'exit `ASNLOAD` (Linux, UNIX, Windows, z/OS)

Outre la personnalisation du code d'exit lui-même, vous pouvez personnaliser le comportement de la routine d'exit `ASNLOAD` via la mise à jour des colonnes de la table `IBMSNAP_SUBS_MEMBR` ou la mise à jour d'un fichier de configuration.

### Utilisation de la table `IBMSNAP_SUBS_MEMBR` pour définir les options `ASNLOAD`

Vous pouvez utiliser les colonnes de la table `IBMSNAP_SUBS_MEMBR` pour personnaliser le comportement de la routine d'exit `ASNLOAD`.

#### A propos de cette tâche

Utilisez la colonne `LOADX_TYPE` pour spécifier une option de chargement. Les valeurs valides de la colonne `LOADX_TYPE` sont les suivantes :

**null (valeur par défaut)**

z/OS

Utiliser l'utilitaire LOAD from CURSOR.

Linux UNIX Windows

La routine d'exit ASNLOAD détermine l'utilitaire le plus approprié (option 3, 4 ou 5).

- 1 Ne pas appeler la routine d'exit ASNLOAD pour ce membre.  
Affectez la valeur 1 à LOADX\_TYPE si vous ne souhaitez pas que la routine d'exit ASNLOAD soit appelée pour ce membre.
- 2 Fournir votre propre logique d'exit.  
Si vous souhaitez inclure votre propre logique dans la routine d'exit ASNLOAD, affectez la valeur 2 à LOADX\_TYPE pour les membres d'ensembles d'abonnements devant être remplis par la routine d'exit ASNLOAD. Si vous affectez la valeur 2 à LOADX\_TYPE sans indiquer de logique d'exit, la routine d'exit échoue.
- 3 Utiliser l'utilitaire LOAD from CURSOR.

Linux UNIX Windows

La fonction LOAD from CURSOR exige qu'une instruction SELECT effectue l'extraction des données à charger dans la table cible (celle-ci doit se trouver dans une base de données locale). Cette instruction peut faire référence à une table DB2 ou à un pseudonyme, et la configuration doit être la suivante :

Si vous effectuez la réplication à partir d'une source non IBM vers une table DB2 dans laquelle le pseudonyme de source enregistrée se trouve dans une autre base de données que la base de données cible ou si vous effectuez la réplication à partir d'une table DB2 vers une autre table DB2 et que la base de données source est différente de la base de données cible, vous devez effectuer les opérations suivantes :

1. Créez un pseudonyme pour les tables source dans la base de données du serveur cible.
2. Mettez à jour le nom du propriétaire du pseudonyme et les colonnes de noms de tables (LOADX\_SRC\_N\_OWNER et LOADX\_SRC\_N\_TABLE) de la table IBMSNAP\_SUBS\_MEMBR.

Si vous effectuez la réplication à partir d'une table DB2 vers une autre table DB2 et que les bases de données source et cible sont les mêmes, ou encore si vous effectuez la réplication à partir d'une source non IBM vers une table DB2 dans laquelle le pseudonyme de source enregistrée se trouve dans la même base de données que la base de données cible, aucune opération supplémentaire n'est à effectuer pour pouvoir utiliser l'utilitaire LOAD from CURSOR.

Linux UNIX Windows 4

Utiliser une combinaison de l'utilitaire EXPORT et de l'utilitaire LOAD.

Linux UNIX Windows 5

Utiliser une combinaison de l'utilitaire EXPORT et de l'utilitaire IMPORT.

### Utilisation du fichier de configuration pour ASNLOAD (Linux, UNIX, Windows)

Vous pouvez utiliser un fichier de configuration afin de configurer les entrées de la routine d'exit ASNLOAD. Ce fichier n'est pas nécessaire pour l'exécution de la routine ASNLOAD.

#### A propos de cette tâche



Le fichier de configuration doit s'appeler `asnload.ini`. La routine d'exit ASNLOAD recherche ce fichier de configuration facultatif dans le répertoire indiqué par le paramètre `apply_path`.

### Procédure

Pour utiliser le fichier de configuration ASNLOAD, procédez comme suit :

1. Modifiez l'exemple de fichier `sqlib/samples/repl/asnload.ini`.
2. Enregistrez ce fichier dans le répertoire indiqué par le paramètre `apply_path` pour l'instance Apply qui a appelé la routine d'exit ASNLOAD.

## Régénération des tables cible avec la routine d'exit ASNLOAD (System i)

Vous pouvez utiliser la routine d'exit ASNLOAD pour régénérer les tables cible de manière plus efficace sur System i. Vous pouvez également modifier la routine avant de l'utiliser.

### Avant de commencer

- Les colonnes de tables cible doivent correspondre à l'ordre et au type de données des tables source.
- La table cible ne peut contenir que des colonnes qui font partie du mappage de réplication.

### A propos de cette tâche

Par exemple, si vous copiez chaque ligne et chaque colonne d'une table source dans une table cible, vous pouvez concevoir une routine d'exit à régénération intégrale qui utilise un fichier DDM (Distributed Data Management) et une commande CL Copy File (**CPYF**) pour copier tout le fichier de la table source dans la table cible.

Pour utiliser la routine d'exit ASNLOAD comme indiqué, démarrez le programme Apply en utilisant le paramètre `FULLREFPGM`.

### Procédure

Pour utiliser une version modifiée de la routine d'exit ASNLOAD, procédez comme suit :

1. Modifiez la routine d'exit ASNLOAD pour qu'elle réponde aux besoins de votre site. Consultez la section PROLOG du programme exemple pour obtenir des informations sur la manière de modifier cette routine d'exit. La source est disponible dans les langages C, COBOL, et RPG, comme indiqué dans le tableau 12.

Tableau 12. Code source pour ASNLOAD

Langage du compilateur	Nom de bibliothèque	Nom de fichier source	Nom de membre
C	QDP4	QCSRC	ASNLOAD
COBOL	QDP4	QCBLESRC	ASNLOAD
RPG	QDP4	QRPGLESRC	ASNLOAD

2. Compilez et effectuez la liaison du programme, puis insérez le fichier exécutable dans le répertoire approprié. Pour éviter des interférences avec le

programme Apply, compilez la routine d'exit afin qu'elle utilise un nouveau groupe d'activation (et non le groupe d'activation de l'appelant).

Vous pouvez compiler la routine d'exit avec un groupe d'activation nommé ou avec un nouveau groupe d'activation. Pour obtenir de meilleures performances, utilisez un groupe d'activation nommé. Avec un groupe d'activation nommé, la routine d'exit doit valider ou annuler les modifications, selon les besoins. Le programme Apply ne provoquera pas la validation ou l'annulation des modification (à moins qu'il ne se termine). La routine d'exit doit explicitement valider les modifications, ou elle doit être compilée pour valider implicitement les modification lorsqu'elle se termine. Toute modification non validée lorsque la routine d'exit se termine n'est pas validée tant que :

- Le programme Apply n'appelle pas une autre routine d'exit avec le même groupe d'activation.
  - Le travail lancé pour le programme Apply n'est pas terminé.
3. Lancez le programme Apply avec le paramètre FULLREFPGM défini sur le nom du programme ASNLOAD. Lorsque vous lancez le programme Apply, il utilise la routine d'exit ASNLOAD que vous avez spécifiée. Si vous voulez qu'il utilise une autre routine d'exit ASNLOAD, arrêtez le programme Apply et relancez-le.

Lorsque vous exécutez la routine d'exit ASNLOAD, elle régénère toutes les tables cible, l'une après l'autre.

## Actualisation d'une table dans un ensemble d'abonnements à plusieurs tables

Vous pouvez demander aux programmes de réplication de recharger une table dans un ensemble d'abonnements contenant plusieurs tables. Pour cela, utilisez le langage SQL pour mettre à jour les valeurs dans la table IBMSNAP\_SUBS\_MEMBR.

Pour actualiser une seule table dans un ensemble de plusieurs tables, exécutez l'instruction de mise à jour SQL suivante sur le serveur de contrôle Apply. L'instruction SQL réinitialise la valeur de la colonne MEMBER\_STATE avec la valeur N, ce qui entraîne une actualisation complète de la table.

```
UPDATE ASN.IBMSNAP_SUBS_MEMBR
SET MEMBER_STATE='N'
WHERE APPLY_QUAL= qual_apply AND
SET_NAME = nom_ensemble AND
WHOS_ON_FIRST = qui_est_en_premier AND
SOURCE_OWNER = propriétaire_source AND
SOURCE_TABLE = table_source AND
TARGET_OWNER = propriétaire_cible AND
TARGET_TABLE = table_cible
```

**Remarque :** Toutes les tables que vous ne voulez pas actualiser doivent avoir la valeur L ou S dans la colonne MEMBER\_STATE.

Lors du cycle suivant, le programme Apply effectue une actualisation complète pour tous les membres ayant une valeur N et remplace la valeur de MEMBER\_STATE par L une fois la table cible chargée. Pour les cycles suivants, l'actualisation différentielle est reprise à partir du point de synchronisation de l'ensemble arrêté pour tous les membres de l'ensemble d'abonnements.

## Veiller à ce que les utilitaires utilisés pour la régénération intégrale attendent les données validées

Si vous indiquez que le programme Apply utilise la routine d'exit ASNLOAD avec une source DB2 pour Linux, UNIX et Windows version 9.7 et ultérieure, assurez-vous que les utilitaires utilisés pour la régénération intégrale attendent les données validées.

### A propos de cette tâche


Dès la version 9.7, DB2 utilise par défaut la sémantique *currently committed* (actuellement validé), dans laquelle les applications telles que les utilitaires de chargement qui lisent les données des tables, n'attendent pas la mise en place de verrouillages de lignes par les applications d'écriture. En effet, les applications de lecture renvoient les données basées sur la version actuellement validée, c'est-à-dire les données antérieures au démarrage de l'opération d'écriture.

Dans la mesure où une régénération intégrale de la table cible nécessite les dernières données validées depuis la table source, assurez-vous que les utilitaires attendent que toutes les transactions en cours susceptibles de modifier la table source, soient terminées avant de commencer le chargement. Ce comportement est appelé "attente de résultat".

Si vous indiquez l'utilitaire LOAD from CURSOR pour ASNLOAD, vous devez utiliser une nouvelle option de serveur fédéré, `CONCURRENT_ACCESS_RESOLUTION=W`, pour forcer le comportement d'attente de résultat sur le pseudonyme utilisé pour la régénération intégrale.

**Restriction :** vous pouvez définir cette option uniquement pour un serveur enregistré de type DB2/UDB version 9.7 ou ultérieure.

Prenez en compte les considérations suivantes :

- Les procédures de définition de l'accès simultané de l'utilitaire LOAD from CURSOR diffèrent selon que la version du serveur de contrôle Apply est 9.7 (ou ultérieure) ou une version antérieure à 9.7.
-  Il n'existe actuellement aucune solution pour forcer le comportement d'attente de résultat lorsque le programme Apply sous z/OS utilise LOAD from CURSOR pour effectuer la régénération intégrale depuis une base de données source DB2 V9.7. Dans ce cas, la meilleure solution consiste à interrompre toutes les applications mettant à jour la table source, dès l'activation de l'enregistrement et jusqu'au démarrage de la régénération intégrale.

### Procédure

Pour veiller à ce que les utilitaires utilisés pour la régénération intégrale attendent les données validées, utilisez l'une des procédures suivantes en fonction de l'utilitaire :

Utilitaire utilisé avec ASNLOAD	Procédure.
LOAD from CURSOR	<p><b>Apply, version 9.7 ou ultérieure</b>  Emettez la commande suivante sur le serveur de contrôle Apply :</p> <pre>db2 alter server <i>nom_serveur</i> OPTIONS (ADD CONCURRENT_ACCESS_RESOLUTION 'W');</pre> <p>Cette commande est obligatoire même s'il s'agit de la version 9.7 de DB2 ou d'une version ultérieure.</p> <p><b>Apply, version antérieure à 9.7</b>  <b>Remarque :</b> 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"> <li>1. Depuis le serveur de contrôle Apply, connectez-vous à la base de données source.</li> <li>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> </li> <li>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> </li> </ol> <p><b>Recommandation :</b> 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 <code>CONCURRENT_ACCESS_RESOLUTION=W</code> et en permettant à d'autres applications d'utiliser le nom de serveur existant.</p>
EXPORT/IMPORT EXPORT/LOAD	<p>Si vous prévoyez d'utiliser l'utilitaire DB2 EXPORT pour actualiser des tables cible à partir d'une source DB2 version 9.7 ou ultérieure et que l'ID utilisateur qui démarre le programme Apply ne dispose pas des droits BINDADD, vous devez effectuer la liaison suivante avant le démarrage de Apply :</p> <pre>db2 bind @db2ubind.lst CONCURRENTACCESSRESOLUTION WAIT FOR OUTCOME COLLECTION ASN</pre>

---

## Chapitre 11. Utilisation des programmes de réplication (z/OS)

Les rubriques suivantes décrivent l'utilisation des programmes de réplication sur le système d'exploitation z/OS.

---

### Utilisation de tâches démarrées par le système pour utiliser les programmes de réplication

z/OS

Vous pouvez utiliser les tâches démarrées par le système pour utiliser le programme Capture, le programme Apply, et le moniteur d'alertes de réplication.

#### Procédure

Pour utiliser les tâches démarrées par le système pour faire fonctionner les programmes de réplication, utilisez cet exemple du programme Capture :

1. Créez une procédure *nomprocédure* dans votre PROCLIB.
2. Créez une entrée dans la classe RACF STARTED pour le *nomprocédure*. Cette entrée associe le *nomprocédure* avec l'ID utilisateur RACF qui doit être utilisé pour démarrer le programme Capture. Veillez à ce que l'autorisation DB2 nécessaire soit accordée à cet ID utilisateur avant de démarrer le programme Capture.
3. A partir de la console système MVS, exécutez la commande **start** *nomprocédure*.

L'exemple de procédure suivant est destiné au programme Capture :

```
//CAPJAYC PROC
//ASNCAP EXEC PGM=ASNCAP,REGION=M,
//PARM='V71A autostop LOGSTDOUT startmode=COLD
//capture_schema=JAY logreuse'
//STEPLIB DD DISP=SHR,DSN=DPROPR.ASN81 .SASNLOAD
//DD DISP=SHR,DSN=SYS1.SCEERUN
//DD DISP=SHR,DSN=DSN7.SDSNLOAD
//CEEDUMP DD SYSOUT=
//SYSPRINT DD SYSOUT=
//SYSTEM DD DUMMY
//
```

---

### Utilisation du langage JCL pour faire fonctionner les programmes de réplication

Sur z/OS, vous pouvez utiliser le langage JCL pour démarrer, arrêter et modifier les programmes de réplication en cours d'exécution. Cela vous permet d'enregistrer des scripts si vous exécutez l'opération de manière répétée.

#### A propos de cette tâche

La bibliothèque d'échantillons de réplication SQL contient des échantillons de langage JCL et des scripts.

**Recommandation :** Copiez les travaux de la bibliothèque SASNSAMP dans une autre bibliothèque avant de faire toute modification. Voir le Répertoire de programmes pour obtenir une liste complète des exemples de travaux que l'on trouve dans la bibliothèque SASNSAMP.

### Procédure

Pour utiliser le programme de réplication avec le langage JCL :

1. Démarrez les programmes de réplication.

Option	Description
Démarrez le programme Capture avec un travail par lots.	Préparez le langage JCL pour z/OS en indiquant les paramètres d'appel optionnels appropriés dans la zone PARM du travail par lots ASNSTRC. Exécutez le travail à partir de TSO ou de la console z/OS. Vous pouvez trouver le travail dans la bibliothèque d'échantillons SASNSAMP.
Démarrez le programme Apply avec un travail par lots.	Préparez le langage JCL pour z/OS en indiquant les paramètres d'appel optionnels appropriés dans la zone PARM du travail par lots ASNSTRA. Exécutez le travail à partir de TSO ou de la console z/OS. Vous pouvez trouver le travail dans la bibliothèque d'échantillons SASNSAMP.
Démarrez le moniteur d'alertes de réplication avec un travail par lots.	Préparez le langage JCL pour z/OS en indiquant les paramètres d'appel optionnels appropriés dans la zone PARM du travail par lots ASNSTRM. Exécutez le travail à partir de TSO ou de la console z/OS. Vous pouvez trouver le travail dans la bibliothèque d'échantillons SASNSAMP.
Démarrez le moniteur d'alertes de réplication avec le langage JCL.	Préparez le langage JCL pour z/OS en indiquant les paramètres d'appel appropriés dans la zone PARM du travail moniteur d'alertes de réplication. Personnalisez le langage JCL afin de satisfaire aux exigences de votre site. Un exemple de langage JCL d'appel dans la bibliothèque SASNSAMP(ASNMON#) est inclus avec le moniteur d'alertes de réplication pour z/OS.  Un exemple de cette ligne dans le langage JCL d'appel est : <pre>//monasn EXEC PGM=ASNMON,PARM='monitor_server=DSN monitor_qual=monqual '</pre> où DSN est un nom de sous-système et monqual est le qualificatif du moniteur.

2. Facultatif : Modifiez les programmes de réplication qui ont déjà démarré.

Après avoir démarré le programme Capture, le programme Apply ou le programme du moniteur d'alertes de réplication, vous pouvez utiliser la commande MODIFY pour arrêter le programme ou effectuer des tâches associées. Vous devez exécuter la commande MODIFY à partir d'une console MVS. Vous pouvez utiliser l'abréviation F, comme indiqué dans l'exemple de syntaxe ci-dessous :

►►—F—*nom de travail*—, —| Paramètres |—————►►

A la base, F *nom de travail*, remplace le nom de commande réel : **asnacmd**, **asnccmd**, ou **asnmcmd**. Par exemple, pour arrêter le programme Capture, vous devez utiliser la commande suivante :

F capjfa,stop

Pour plus d'informations sur la commande FMODIFY, voir les commandes système z/OS MVS.

---

## Démarrage du programme Apply sur z/OS avec JCL

Vous pouvez démarrer le programme Apply sur z/OS en modifiant et en exécutant un échantillon de script rédigé extrait de votre répertoire d'échantillons.

### Procédure

Pour démarrer le programme Apply sur z/OS avec JCL :

1. Préparez le langage JCL pour z/OS en indiquant les paramètres d'appel appropriés dans la zone PARM du travail Apply.
2. Personnalisez le langage JCL afin qu'il soit conforme aux exigences de votre site.

Pour les systèmes d'exploitation z/OS, un exemple de cette ligne dans le langage JCL d'appel est :

```
//apasn EXEC PGM=ASNAPPLY,PARM='control_server=CTLDB1
                                DB2_SUBSYSTEM=DSN
                                apply_qual=myqual spillfile=disk'
```

Pour les systèmes d'exploitation UNIX et Window, un exemple de cette ligne dans le langage JCL d'appel est :

```
//apasn EXEC PGM=ASNAPPLY,PARM='control_server=CTLDB1
                                apply_qual=myqual spillfile=disk'
```

3. Soumettre le JCL à partir de la TSO ou de la console MVS.

---

## Gérer les programmes de réplication SQL en cours d'exécution grâce à l'utilisation de la commande MVS MODIFY

Après avoir démarré le programme Capture, le programme Apply ou le moniteur d'alertes de réplication, vous pouvez utiliser la commande **MODIFY** pour arrêter le programme ou effectuer des tâches associées.

### 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 nom de travail* , remplace le nom de commande réel : **asnccmd**, **asnacmd**, 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 Apply qui utilise le travail PLS, vous devez lancer la commande suivante :

```
F PLS,stop
```

Le tableau 13 dresse la liste des commandes Capture que vous pouvez exécuter avec le mot clé **f**. Dans tous les exemples, le nom de travail est mycap.

Tableau 13. Exemple de commandes MODIFY pour le programme Capture

Paramètre	Exemple de commande utilisant le mot clé f
prune	f mycap,prune

Tableau 13. Exemple de commandes MODIFY pour le programme Capture (suite)

Paramètre	Exemple de commande utilisant le mot clé f
<b>qrypargs</b>	f mycap,qrypargs
<b>reinit</b>	f mycap,reinit
<b>suspend</b>	f mycap,suspend
<b>resume</b>	f mycap,resume
<b>status</b>	f mycap,status
<b>stop</b>	f mycap,stop
<b>chgpargs</b>	f mycap,chgpargs autostop=y f mycap,chgpargs commit_interval=n f mycap,chgpargs logreuse=y f mycap,chgpargs logstdout=y f mycap,chgpargs memory_limit=n f mycap,chgpargs monitor_interval=n f mycap,chgpargs monitor_limit=n f mycap,chgpargs prune_interval=n f mycap,chgpargs retention_limit=n f mycap,chgpargs signal_limit=n f mycap,chgpargs sleep_interval=n f mycap,chgpargs term=y f mycap,chgpargs trace_limit=n

Le tableau 14 dresse la liste des commandes Apply que vous pouvez exécuter avec le mot clé f. Dans tous les exemples, le nom de travail est myapp.

Tableau 14. Exemples de commandes MODIFY pour le programme Apply

Paramètre	Exemple de commande utilisant le mot clé f
<b>status</b>	f myapp,status
<b>stop</b>	f myapp,stop

Le tableau 15 dresse la liste des commandes du programme asntrc que vous pouvez exécuter avec le mot clé f. Dans tous les exemples, le nom de travail est mycap.

Tableau 15. 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 <b>asntrc</b>	f mycap,asntrc on f mycap,asntrc statlong
Formatez un rapport asntrc fmt et envoyez la sortie à un ensemble de données z/OS	F mycap, asntrc fmt -ofn // 'USRT001.TRCFMT'
Formatez un rapport asntrc flw et envoyez la sortie à un ensemble de données z/OS	F mycap, asntrc flw -ofn // 'USRT001.TRCFLW'
Arrêtez une trace de programme	F mycap, 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 16 dresse la liste des 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 16. Exemples de commandes MODIFY pour le programme du moniteur d'alertes de réplication

Paramètre	Exemple de commande utilisant le mot clé f
<b>reinit</b>	f mymon,reinit
<b>status</b>	f mymon,status
<b>qryparms</b>	f mymon,qryparms
<b>suspend</b>	f mymon,suspend
<b>resume</b>	f mymon,resume
<b>stop</b>	f mymon,stop
<b>chgparms</b>	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.

---

## Démarrage du programme Capture sur z/OS avec JCL

Vous pouvez démarrer le programme Capture sur z/OS en modifiant et en exécutant un échantillon de script rédigé extrait de votre répertoire d'échantillons.

### Procédure

Pour démarrer le programme Capture sur z/OS avec JCL :

1. Préparez le langage JCL pour z/OS.
  - a. Indiquez les paramètres d'appel optionnels appropriés dans la zone PARM du travail par lots Capture.
  - b. Si vous n'avez pas paramétré la variable d'environnement TZ dans le fichier système /etc/profile ou dans le fichier .profile dans le répertoire de base de l'utilisateur exécutant le programme de réplication, vous devez paramétrer les variables TZ et d'environnement de langage dans le langage JCL. Pour plus d'informations sur la définition de la variable TZ, voir Définition du fuseau horaire.

L'exemple suivant de cette ligne dans le langage JCL d'appel comprend le paramétrage des variables TZ et LANG :

```
//CAPJFA EXEC PGM=ASNCAP, PARM='ENVAR('TZ=PST8PDT','LANG=en_US')/  
DSN6 cold capture_schema=JFA autostop'
```

2. Soumettre le JCL à partir de la TSO ou de la console MVS.

---

## 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	ASNQCxxxxxyyyy
Q Apply	ASNQAxxxxxyyyy
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 concaténera 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.

---

## Migration de votre environnement de réplication en mode de partage de données (z/OS)

Si le programme Capture s'exécute en mode de non partage des données et que vous migrez votre installation en mode de partage de données, vous devez préparer vos systèmes pour qu'ils s'exécutent dans un Sysplex en exécutant une seule fois l'utilitaire **ASNPLXFY**.

### Avant de commencer

Utilisez le même ID utilisateur que vous utilisez pour exécuter le programme Capture, ou un ID utilisateur ayant les mêmes droits d'accès. Veillez à ce que l'utilitaire **ASNPLXFY** soit autorisé par APF. Le plan ASNPLXFY doit être associé au sous-système. De même, le sous-système doit s'exécuter en mode de partage des données. Pour plus de détails concernant l'association de cet utilitaire, voir le Répertoire de programmes.

### A propos de cette tâche

Exécutez cet utilitaire dans la configuration de partage des données avant le démarrage à chaud du programme Capture, afin que celui-ci démarre au LRSN adéquat. Cet utilitaire migre les données dans la table IBMSNAP\_RESTART. Il

convertit les numéros de séquence de journal ne partageant pas les données (RBA) en numéros de séquence équivalents (LRSN) dans un environnement de partage des données.

### **Procédure**

Pour exécuter l'utilitaire **ASNPLXFY** dans l'environnement de partage des données USS :

1. Arrêtez le programme Capture
2. Emettez la commande **ASNPLXFY** à partir d'une ligne de commande. Voici un exemple :

```
ASNPLXFY votresous-système schémacapture
```

où le nom du sous-système est requis et le schéma Capture est optionnel. Le schéma Capture par défaut est ASN.

3. Effectuez un démarrage à chaud du programme Capture.

---

## Chapitre 12. Modification d'un environnement de réplication SQL

Les rubriques suivantes décrivent les problèmes et les procédures relatives aux modifications quotidiennes d'un environnement de réplication Q.

---

### Enregistrement de nouveaux objets

Vous pouvez à tout moment enregistrer une nouvelle table, une nouvelle vue ou un nouvel pseudonyme au sein de votre environnement de réplication. Pour cela, il est inutile de réinitialiser le programme Capture.

#### A propos de cette tâche

Un nouvel objet enregistré est automatiquement initialisé par le programme Capture lorsque le programme Apply effectue pour la première fois le traitement d'un ensemble d'abonnements se rapportant à cet objet. Le programme Apply signale au programme Capture qu'il doit commencer à capturer les modifications pour ce nouvel objet.

#### Procédure

Pour enregistrer de nouveaux objets, procédez comme suit :

Pour enregistrer de nouveaux objets, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Programme de ligne de commande ASNCLP</b>	Utilisez la commande <b>CREATE REGISTRATION</b> pour enregistrer une table source, une vue ou un pseudonyme. Par exemple, les commandes suivantes définissent l'environnement et enregistrent la table DEPARTMENT dans la base de données SAMPLE DB2 pour la réplication avec régénération intégrale. <pre>SET SERVER CAPTURE TO DB SAMPLE; SET OUTPUT CAPTURE SCRIPT "registernew.sql"; SET LOG "registernew.err"; SET RUN SCRIPT LATER; CREATE REGISTRATION (DB2ADMIN.DEPARTMENT) FULL REFRESH ONLY;</pre>
<b>Centre de réplication</b>	Utilisez l'une des fenêtres suivantes : <ul style="list-style-type: none"><li>• Bloc-notes des Propriétés des tables enregistrées</li><li>• Bloc-notes des Propriétés des vues enregistrées</li><li>• Bloc-notes des Propriétés des pseudonymes enregistrés</li></ul> Pour ouvrir ces fenêtres, cliquez dans l'arborescence des objets sur le dossier <b>Tables enregistrées</b> , <b>Vues enregistrées</b> ou <b>Pseudonymes enregistrés</b> sous un serveur de contrôle Capture ; ensuite, cliquez avec le bouton droit sur l'objet enregistré souhaité dans le panneau de contenu et sélectionnez <b>Propriétés</b> .
<b>System i</b> Commande système <b>ADDDPRREG</b>	Utilisez la commande d'ajout d'enregistrement (ADDDPRREG) pour enregistrer une nouvelle table sous System i.

---

## Modification des attributs d'enregistrement d'objets enregistrés

Vous pouvez à tout moment modifier les attributs d'enregistrement d'objets enregistrés existants.

### Procédure

Pour modifier les attributs d'enregistrement d'objets enregistrés, procédez comme suit :

1. Modifiez les attributs en utilisant l'une des méthodes suivantes :

Méthode	Description
<b>Programme de ligne de commande ASNCLP</b>	Utilisez la commande ALTER REGISTRATION pour modifier les propriétés d'un objet enregistré. Par exemple, les commandes suivantes définissent l'environnement et modifient l'enregistrement de la table STAFF dans la base de données DB2 SAMPLE, de telle sorte que les mises à jour soient capturées en tant que couples de suppressions-insertions : <pre>SET SERVER CAPTURE TO DB SAMPLE; SET OUTPUT CAPTURE SCRIPT "register.sql"; SET LOG "register.err"; SET RUN SCRIPT LATER; ALTER REGISTRATION (DB2ADMIN.STAFF) UPDATE AS DELETE INSERT ON;</pre>
<b>Centre de réplication</b>	Utilisez l'une des fenêtres suivantes : <ul style="list-style-type: none"><li>• Bloc-notes des Propriétés des tables enregistrées</li><li>• Bloc-notes des Propriétés des vues enregistrées</li><li>• Bloc-notes des Propriétés des pseudonymes enregistrés</li></ul> Pour ouvrir ces fenêtres, cliquez dans l'arborescence des objets sur le dossier <b>Tables enregistrées</b> , <b>Vues enregistrées</b> ou <b>Pseudonymes enregistrés</b> sous un serveur de contrôle Capture ; ensuite, cliquez avec le bouton droit sur l'objet enregistré souhaité dans le panneau de contenu et sélectionnez <b>Propriétés</b> .

2. Une fois les attributs modifiés, réinitialisez le programme Capture.

---

## Ajout de colonnes aux tables source

Si vous devez ajouter des colonnes à une table source enregistrée, déterminez tout d'abord la façon dont la réplication DB2 utilise cette table. Si vous devez répliquer les nouvelles colonnes dans cette table source, vous devez vous assurer que les programmes Capture et Apply existants reconnaissent les nouvelles colonnes et poursuivent le traitement en cours sans interruption.

### Avant de commencer

Avant d'exécuter cette procédure, familiarisez-vous avec les structures des tables source, des tables de modification des données (CD) et des tables cible ainsi et des enregistrements et ensembles d'abonnements définis sur votre système.

### Restrictions

- **System i** La modification d'une table source enregistrée sous System i pour ajouter une nouvelle colonne n'est pas prise en charge. System i crée une entrée de journal EJ (end journaling) avant de modifier la table source lors d'une opération ALTER. Lorsque vous ajoutez une nouvelle colonne à une table source,

vous devez supprimer et recréer l'enregistrement et l'abonnement de la table. Lorsque vous ajoutez des colonnes à une table System i utilisant un numéro relatif d'enregistrement comme clé principale, supprimez l'enregistrement, ajoutez la colonne à la table source puis ajoutez de nouveau cette table en tant que nouvel enregistrement. Indiquez que le numéro relatif d'enregistrement sera enregistré.

- Vous ne pouvez pas utiliser cette procédure pour ajouter des colonnes à des sources enregistrées situées dans des bases de données relationnelles non DB2. Un enregistrement de source relationnelle non DB2 inclut un ensemble de déclencheurs utilisés pour la capture des modifications. Vous ne pouvez pas modifier ces déclencheurs. Par conséquent, si vous devez ajouter de nouvelles colonnes à cette table source et si vous devez répliquer les données dans ces colonnes, vous devez supprimer puis créer de nouveau la source enregistrée existante.

### A propos de cette tâche

Vous aurez peut-être besoin d'exécuter certaines procédures, en fonction de la sélection ou non de la répllication de données dans les nouvelles colonnes.

#### Non répliquées

Si vous ne souhaitez pas répliquer les données dans les nouvelles colonnes, il est inutile d'exécuter une procédure spécifique. Le programme Capture reconnaît immédiatement les modifications apportées et continue de fonctionner.

#### Répliquées

Si vous souhaitez répliquer les données dans ces nouvelles colonnes, exécutez la procédure suivante pour vous assurer que les données de ces nouvelles colonnes sont capturées et que les programmes Capture et Apply continuent de fonctionner sans erreurs.

### Procédure

Pour ajouter des colonnes aux tables source, procédez comme suit :

1. Mettez au repos l'activité relative à la table source à modifier.
2. Arrêtez le programme Capture
3. Facultatif : Si vous avez besoin que le programme Capture reste actif pendant l'exécution de cette procédure, insérez un signal USER dans la table IBMSNAP\_SIGNAL après avoir arrêté l'activité relative à la table source. Attendez que le programme Capture traite le signal USER. Une fois que le programme Capture a traité le signal USER, il n'a plus d'activité à traiter pour la table de modification des données (CD) associée ; il n'a donc plus besoin d'accéder à cette table.
4. A partir du Centre de répllication, désactivez tous les ensembles d'abonnements qui sont abonnés à cette table source.

**Remarque :** Si vous ne souhaitez pas désactiver les ensembles d'abonnements au cours de ce processus, vérifiez qu'aucun programme Apply associé à ces ensembles n'est en cours de fonctionnement pour la table source lorsque vous ajoutez les nouvelles colonnes. Vous pouvez également vérifier que ces programmes Apply ont traité des données jusqu'au numéro LSN associé au signal USER précédent.

Les méthodes décrites dans cette étape garantissent un accès exclusif à la table de modification des données (CD), ce qui vous permet de modifier la table.

5. Utilisez une instruction ALTER TABLE ADD dans SQL pour ajouter les nouvelles colonnes à la table source.
6. Ajoutez les nouvelles colonnes à la table de modification des données (CD) ; pour cela, utilisez la commande ALTER REGISTRATION dans le programme de ligne de commande ASNCLP ou dans le bloc-notes Propriétés des tables enregistrées, dans le Centre de réplication. Le programme Capture réinitialise automatiquement l'enregistrement et capture les modifications apportées à ces nouvelles colonnes lors de la lecture initiale des données de journal.
7. Utilisez une instruction ALTER TABLE ADD dans SQL pour ajouter les nouvelles colonnes à la table cible.
8. Désactivez tous les ensembles d'abonnements associés qui n'ont pas été désactivés à partir du Centre de réplication. En cas de besoin absolu, vous pouvez désormais reprendre l'activité relative à cette table source. Cependant, puisque les ensembles d'abonnements associés n'ont pas encore été modifiés, ces ensembles doivent rester désactivés afin que vous ne perdiez pas certaines des modifications apportés à ces nouvelles colonnes.
9. Ajoutez les nouvelles colonnes aux membres d'ensembles d'abonnements associés ; pour cela, utilisez la commande ALTER MEMBER ADD COLS dans le programme de ligne de commande ASNCLP ou dans la fenêtre Ajouter une colonne à la table cible, dans le Centre de réplication.
10. Facultatif : Si l'une des colonnes ajoutées contient des valeurs par défaut, exécutez l'utilitaire REORG sur la table source.
11. z/OS Linux UNIX Windows Si vous exécutez le programme Apply et que le paramètre **opt4one** porte la valeur y, arrêtez puis redémarrez le programme Apply.
12. Réactivez les ensembles d'abonnements.

---

## Traitement des opérations ALTER TABLE ALTER COLUMN SET DATA TYPE

A partir de la version 10.1 sur z/OS et Linux, UNIX et Windows, la Réplication SQL traite les opérations ALTER TABLE ALTER COLUMN SET DATA TYPE à la table source en modifiant le type de données de la colonne de la table CD correspondante. Vous devez tout de même modifier la colonne de la table cible.

### Conditions requises :

- La version 10.1 ou une version plus récente du serveur Capture doit être installée sous z/OS, Linux, UNIX et Windows. L'exception étend une colonne VARCHAR/VARGRAPHIC. Cette opération est prise en charge sur des versions plus anciennes.
- 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).

Capture applique également le nouveau type de données à toutes les colonnes d'image-avant dans la table de modification des données.

z/OS Sous z/OS, après modification de la table CD, DB2 place la table dans l'état REORG PENDING, qui nécessite une opération REORG avant de pouvoir utiliser la table. Le programme Capture appelle la procédure stockée DB2



ADMIN\_REVALIDATE\_DB\_OBJECTS pour retirer la table de l'état REORG PENDING. Les exceptions à cette condition REORG s'étendent à la longueur des colonnes VARCHAR ou VARGRAPHIC.

---

## Arrêt de la capture des modifications des objets enregistrés

Vous devez désactiver un objet enregistré avant de le supprimer, pour vous assurer que le programme Capture termine tous les traitements appliqués à l'objet. Vous pouvez par ailleurs désactiver un objet enregistré si vous souhaitez arrêter temporairement la capture des modifications pour cet objet, mais que vous devez conserver vos programmes Capture en cours d'exécution pour d'autres objets enregistrés.

### Restrictions

Vous pouvez désactiver uniquement les objets enregistrés DB2 qui sont définis en tant que sources Capture.

Vous ne pouvez pas désactiver des objets de base de données relationnelle non DB2 utilisés par les déclencheurs Capture.

### A propos de cette tâche

Le programme Capture arrête de capturer les modifications pour les objets source qui ont été désactivés ; toutefois, les tables de modification des données (CD), les attributs d'enregistrement et les ensembles d'abonnements associés à ces objets source restent sur le système.

Avant de désactiver un objet enregistré, vous devez désactiver tous les ensembles d'abonnements associés à cet objet enregistré. Cela garantit que vos programmes Apply ne viendront pas interférer avec le processus de désactivation via la réactivation automatique de l'objet avant sa suppression, ou avant que vous soyez prêt à le réactiver.

Tous les ensembles d'abonnements associés à l'objet enregistré sont affectés lors de la désactivation de l'objet et lorsque la réplication SQL arrête la capture des modifications pour cet objet. Si vous souhaitez poursuivre l'exécution de ces ensembles d'abonnements, vous devez supprimer les membres de ces ensembles qui utilisent cet objet enregistré en tant que source.

### Procédure

Pour désactiver un objet enregistré, procédez comme suit :

1. Désactivez tous les ensembles d'abonnements associés en utilisant le Centre de réplication. Cliquez sur le dossier **Ensembles d'abonnements**, puis cliquez avec le bouton droit de la souris sur les ensembles d'abonnements actifs dans le panneau de contenu et sélectionnez **Désactiver**.
2. Désactivez les objets enregistrés en utilisant l'une des méthodes suivantes :

Méthode	Description
Centre de réplication	Cliquez sur le dossier <b>Tables enregistrées</b> , puis cliquez avec le bouton droit de la souris sur la table enregistrée souhaitée dans le panneau de contenu et sélectionnez <b>Arrêter la capture des modifications</b> .

Méthode	Description
SQL	Insérez manuellement un signal CAPSTOP dans la table IBMSNAP_SIGNAL.

## Enregistrements admissibles pour la réactivation

Lorsque vous réactivez un enregistrement, le programme Capture le réactive une fois que le programme Apply lui a envoyé un signal CAPSTART. Si, cependant, le programme Capture désactive un enregistrement car une erreur imprévue s'est produite, vous devez effectuer une opération spécifique pour réactiver l'enregistrement.

### Avant de commencer

Lisez les messages d'erreur générés par le programme Capture relatifs aux enregistrements désactivés.

Familiarisez-vous avec la structure des tables de contrôle Capture et avec les programmes exécutés sur votre système.

### A propos de cette tâche

Des erreurs imprévues peuvent entraîner la définition, par le programme Capture, de la valeur S (Arrêté) pour la colonne STATE de la table IBMSNAP\_REGISTER si la valeur de la colonne STOP\_ON\_ERROR porte la valeur N pour cet enregistrement. La valeur de la colonne STATE indique que le programme Capture a arrêté de traiter cet enregistrement et qu'il doit être réparé. Le programme Apply n'émet pas de signal CAPSTART pour les enregistrements qui se trouvent à l'état Arrêté.

### Procédure

Pour corriger les erreurs imprévues et rendre les enregistrements admissibles pour la réactivation, procédez comme suit :

1. Modifiez l'enregistrement en utilisant les informations contenues dans les messages d'erreur.
2. Sur le serveur de contrôle Capture, exécutez le script SQL suivant pour réinitialiser la colonne STATE de la table IBMSNAP\_REGISTER.

```
UPDATE schéma.IBMSNAP_REGISTER
   SET STATE = 'I'
 WHERE
   SOURCE_OWNER = 'Schémasource' AND
   SOURCE_TABLE = 'Tablesourc' AND
   SOURCE_VIEW_QUAL = qual_vue_source AND
   STATE = 'S';
```

Où *schéma* correspond au nom du schéma Capture, *Schémasource* correspond au schéma de la table source enregistrée, *Tablesourc* correspond au nom de la table source enregistrée et *qual\_vue\_source* correspond au qualificatif de vue source pour cette table source. Une fois que la colonne STATE porte la valeur I (Inactive), le programme Capture est prêt à commencer la capture des données, dès qu'un signal CAPSTART est reçu (il est généralement envoyé par le programme Apply).

Supposons que la table source d'un enregistrement actif soit endommagée accidentellement au niveau de DATA CAPTURE NONE (qui devrait être DATA CAPTURE CHANGES). Supposons également que cet enregistrement ait été défini avec le paramétrage STOP\_ON\_ERROR = 'N', ce qui indique que le programme Capture ne s'arrête pas en cas de détection d'erreur. Au redémarrage suivant (ou lors de la réinitialisation suivante) du programme Capture, ce dernier reconnaîtra cette erreur dans la table source et affectera la valeur S (Arrêté) à la colonne STATE de la table IBMSNAP\_REGISTER, pour cet enregistrement. Vous recevrez ensuite un message d'erreur lorsque le programme Apply tentera de traiter l'ensemble d'abonnements correspondant, car l'enregistrement portera l'état d'arrêt. Vous devez :

- Corriger le paramétrage de la table source via SQL, en envoyant l'instruction ALTER TABLE, qui permet de réinitialiser l'option de table à DATA CAPTURE CHANGES.
- Faire passer manuellement l'enregistrement de l'état d'arrêt à l'état d'inactivité, à l'aide du script SQL ci-dessus.

Le programme Apply effectue ensuite une régénération intégrale de l'ensemble d'abonnements.

---

## Suppression d'enregistrements

Si vous supprimez un enregistrement, la réplication SQL supprime l'enregistrement de l'objet, supprime également les tables de modification des données (CD) ou de modification cohérente des données (CCD), ainsi que le pseudonyme d'objet et tous les déclencheurs Capture de sources de base de données relationnelle non DB2. La table ou vue cible réelle reste dans la base de données.

### A propos de cette tâche

**Important :** La désactivation est un processus asynchrone. Vous devez vérifier que le processus de désactivation a pris fin avant de supprimer l'objet.

Si vous effectuez des modifications alors que le programme Capture est en cours de fonctionnement, ces modifications ne seront pas reconnues par le programme Capture tant que vous n'aurez pas réinitialisé ou arrêté puis redémarré le programme Capture.

### Procédure

Pour supprimer des enregistrements en utilisant le Centre de réplication, procédez comme suit :

1. Désactivez l'ensemble d'abonnements auquel appartient l'objet enregistré ou arrêtez le programme Apply. Dans le Centre de réplication, cliquez droit sur l'ensemble d'abonnements actif dans le panneau du contenu et sélectionnez **Désactiver**.
2. Désactivez l'enregistrement à supprimer afin de vous assurer que le programme Capture a terminé le traitement de cet objet ou bien arrêtez le programme Capture. Pour désactiver l'enregistrement, cliquez avec le bouton droit sur l'objet enregistré dans le panneau de contenu et cliquez sur **Arrêtez la capture des modifications**.
3. Supprimez le membre d'un ensemble d'abonnements pour l'objet enregistré. Cliquez avec le bouton droit sur l'abonnement dans l'arborescence de

navigation et cliquez sur **Supprimer membres**. Utilisez ensuite Supprimer membres dans le bloc-notes Ensembles d'abonnements pour supprimer le membre et, le cas échéant, la table cible.

4. Supprimez l'enregistrement. Cliquez avec le bouton droit sur l'objet enregistré dans le panneau de contenu, puis sélectionnez **Supprimer**.
5. Réinitialisez le programme Capture de sorte qu'il sélectionne les modifications. Cliquez avec le bouton droit sur le serveur de contrôle de Capture dans l'arborescence de navigation et cliquez sur **Réinitialiser Capture**.
6. Activez l'ensemble d'abonnements. Cliquez droit sur l'ensemble dans le panneau du contenu et cliquez sur **Activer infiniment**.
7. Démarrez le programme Capture ou le programme Apply, ou bien les deux si vous les avez arrêtés.

**System i** Sur System i, exécutez la commande RMVDPRREG (Remove DPR Registration) afin de supprimer une table source spécifique de la table IBMSNAP\_REGISTER.

---

## Modification des schémas Capture

Vous pouvez modifier un schéma Capture existant.

### Avant de commencer

- Familiarisez-vous avec les tables de contrôle de réplication SQL et avec les ensembles d'abonnements définis sur votre système.
- Déterminez le nouveau nom de schéma Capture.
- Vérifiez que votre serveur de contrôle Capture et tous les serveurs de contrôle Apply associés à ce serveur de contrôle Capture ont été migrés vers la version 8 ou une version ultérieure.

### Restrictions

Vous ne devez pas utiliser cette procédure si votre serveur source est une base de données relationnelle non DB2.

### A propos de cette tâche

**Conseil :** Si vous avez créé des définitions de surveillance ou démarré des programmes du moniteur d'alertes de réplication avec le schéma Capture que vous allez modifier, supprimez ces définitions. Une fois que vous avez modifié le schéma Capture, créez de nouveau les définitions en leur affectant le nouveau nom de schéma Capture. Ensuite, vous pouvez réinitialiser les moniteurs associés à l'aide de la commande système **asnmcmd reinit**. Vous pouvez également arrêter les moniteurs à l'aide de la commande système **asnmcmd stop** puis redémarrer les programmes à l'aide de la commande système **asnmmon**.

### Procédure

Pour modifier les schémas Capture, procédez comme suit :

1. Créez les tables de contrôle destinées au nouveau schéma Capture.
2. Arrêtez le programme Capture
3. Désactivez tous les ensembles d'abonnements associés en utilisant le Centre de réplication.

4. Sur le serveur de contrôle Apply, exécutez l'instruction SQL suivante pour modifier les noms de schémas Capture pour les ensembles d'abonnements associés, avec les tables source appartenant à ce schéma Capture :

```
UPDATE ASN.IBMSNAP_SUBS_SET
  SET CAPTURE_SCHEMA = 'NouveauSchéma'
WHERE
  CAPTURE_SCHEMA = 'SchémaExistant';
```

Où *NouveauSchéma* correspond au nom du nouveau schéma Capture et *SchémaExistant* correspond au nom du schéma Capture modifié.

5. Si vous avez créé des ensembles d'abonnements avec des tables cible (des tables CCD ou des tables réplique, par exemple) enregistrées dans ce schéma Capture, exécutez l'instruction SQL suivante à partir du serveur de contrôle Apply pour modifier le nom de schéma cible de ces ensembles d'abonnements :

```
UPDATE ASN.IBMSNAP_SUBS_SET
  SET TGT_CAPTURE_SCHEMA = 'NouveauSchéma'
WHERE
  TGT_CAPTURE_SCHEMA = 'SchémaExistant';
```

Où *NouveauSchéma* correspond au nom du nouveau schéma Capture et *SchémaExistant* correspond au nom du schéma Capture modifié.

6. A partir du serveur de contrôle Capture, exécutez une instruction SQL permettant de copier les informations actives de chaque table de contrôle Capture existante dans chaque nouvelle table de contrôle Capture correspondante créée au cours de l'étape 1. Par exemple, pour copier les informations actives dans la table IBMSNAP\_REGISTER :

```
INSERT INTO NouveauSchéma.IBMSNAP_REGISTER
  SELECT * FROM
    SchémaExistant.IBMSNAP_REGISTER;
```

Où *NouveauSchéma* correspond au nom du nouveau schéma Capture et *SchémaExistant* correspond au nom du schéma Capture modifié.

Répétez cette étape pour chaque table de contrôle Capture existante, y compris certaines ou l'ensemble des tables suivantes :

- IBMSNAP\_CAPMON
- IBMSNAP\_CAPPARMS
- IBMSNAP\_CAPTRACE
- IBMSNAP\_PRUNCNTL
- IBMSNAP\_PRUNE\_SET
- IBMSNAP\_REG\_EXT (System i uniquement)
- IBMSNAP\_REGISTER
- IBMSNAP\_RESTART
- IBMSNAP\_SIGNAL
- IBMSNAP\_UOW

Il est inutile de répéter cette étape pour IBMSNAP\_CAPENQ (sous UNIX, Windows, z/OS) ou la table de contrôle IBMSNAP\_PRUNE\_LOCK, car il n'y a aucune ligne dans ces tables. Ne changez *pas* les tables de modification des données (CD).

7. Supprimez le schéma existant et ses tables de contrôle Capture, en utilisant le Centre de réplification ou le programme ASNCLP.
8. Redémarrez le programme Capture en utilisant le nouveau nom de schéma.
9. Réactivez les ensembles d'abonnements associés en utilisant le Centre de réplification.

---

## Création de nouveaux ensembles d'abonnements

Vous pouvez à tout moment créer de nouveaux ensembles d'abonnements pour un objet enregistré existant, et leur ajouter des membres.

### Avant de commencer

Avant de créer un nouvel ensemble d'abonnements, vous devez enregistrer les tables ou les vues que vous utiliserez comme sources.

### Restrictions


Si le programme Apply correspondant est actif, n'activez pas le nouvel ensemble d'abonnements tant qu'il n'a pas été totalement défini.

### A propos de cette tâche

Cette procédure concerne l'ajout d'un nouvel ensemble d'abonnements, avec ou sans membres.

### Procédure

Pour créer un nouvel ensemble d'abonnements, utilisez l'une des méthodes suivantes :

Méthode	Description
Programme de ligne de commande ASNCLP	Utilisez la commande CREATE SUBSCRIPTION SET pour créer un ensemble vide.
Centre de réplication	Utilisez le bloc-notes <b>Créer un ensemble d'abonnements</b> pour créer un ensemble et ajouter un membre (ou encore pour créer un ensemble vide).  Pour l'ouvrir, développez le serveur de contrôle Apply sur lequel l'ensemble doit être défini, puis cliquez avec le bouton droit de la souris sur le dossier <b>Ensembles d'abonnements</b> ; ensuite, cliquez sur <b>Créer</b> .
 Commande système ADDDPRSUB	Utilisez la commande d'ajout d'ensemble d'abonnements DPR pour créer un ensemble d'abonnements avec membres ou sans membre.

---

## Ajout de membres d'ensembles d'abonnements à des ensembles existants

Vous pouvez ajouter un ou plusieurs membres (chacun utilisant la même table source) à un ou à plusieurs ensembles d'abonnements. Par exemple, si vous sélectionnez trois ensembles d'abonnements, vous pouvez ajouter un membre à chacun d'entre eux, qui utilisent tous la même source de réplication.

### A propos de cette tâche

Lorsque vous ajoutez un membre à un ensemble d'abonnements, vous insérez des informations sur le nouveau membre dans les tables de contrôle Apply. Dans la plupart des cas, le programme Apply lit ces informations au début du cycle suivant.

Cependant, si vous souhaitez ajouter un membre à un ensemble d'abonnements traité sous Linux, UNIX, Windows, ou z/OS avec l'option OPT4ONE ou avec l'option OPTSNGSET sous System i, vous devez arrêter le programme Apply pour cet ensemble, puis le redémarrer. Si vous traitez un ensemble avec l'option OPT4ONE, le programme Apply lit en mémoire les informations de la table de contrôle pour cet ensemble, ce qui lui évite d'aller lire ces informations dans les tables de contrôle au début de chaque cycle Apply.

Si la table source du membre est enregistrée en vue de la réplication différentielle et que le programme Capture est déjà en cours de fonctionnement, il est inutile d'arrêter ou de réinitialiser le programme Capture avant d'ajouter le membre. Le membre ajouté doit utiliser une table enregistrée comme source ; par conséquent, le programme Capture capture déjà les modifications pour lui.

### Procédure

Pour ajouter de nouveaux membres à des ensembles d'abonnements existants, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Programme de ligne de commande ASNCLP</b>	Utilisez la commande CREATE MEMBER pour ajouter un membre d'ensemble d'abonnements à un ensemble existant.
<b>Centre de réplication</b>	Utilisez le bloc-notes <b>Ajouter un membre à un ensemble d'abonnements</b> .  Pour ouvrir ce bloc-notes, cliquez sur le dossier <b>Tables enregistrées</b> . Dans le panneau de contenu, cliquez avec le bouton droit de la souris sur la table enregistrée à utiliser, puis cliquez sur <b>Ajouter un membre</b> .
<b>System i</b>  Commande système <b>ADDDPRSUBM</b>	Utilisez la commande d'ajout de membre d'ensemble d'abonnements (ADDDPRSUBM) pour ajouter un membre d'ensemble d'abonnements à un ensemble existant.

## Désactivation de membres d'ensembles d'abonnements existants

Si vous souhaitez que le programme Apply ignore un membre d'ensemble d'abonnements défectueux et poursuive le traitement du reste de l'ensemble, vous devez désactiver le membre défectueux.

### A propos de cette tâche

En cas de problème lors de la réplication dans une table de l'ensemble d'abonnements, le programme Apply insère un message d'erreur dans la table IBMSNAP\_APPLYTRAIL et poursuit le traitement des autres membres au cours du cycle Apply.

### Procédure

Pour désactiver un membre d'ensemble d'abonnements, exécutez l'instruction SQL UPDATE suivante :

```
UPDATE ASN.IBMSNAP_SUBS_MEMBR
SET MEMBER_STATE = 'D'
WHERE APPLY_QUAL= qual_apply
      SET_NAME = nom_ensemble
      WHOS_ON_FIRST = qui_est_enpremier
      SOURCE_OWNER = propriétaire_source
      SOURCE_TABLE = table_source
      SOURCE_VIEW_QUAL = qual_vue_source
      TARGET_OWNER = propriétaire_cible
      TARGET_TABLE = table_cible
```

Le programme Apply ne traite pas ce membre tant qu'il n'a pas été réactivé.

---

## Activation de membres d'ensembles d'abonnements dans des ensembles d'abonnements existants

Vous pouvez ajouter ou réactiver des membres désactivés d'ensembles d'abonnements : pour cela, affectez à MEMBER\_STATE la valeur N (nouveau).

### Procédure

Pour réactiver un membre d'ensemble d'abonnements, émettez l'instruction SQL UPDATE suivante :

```
UPDATE ASN.IBMSNAP_SUBS_MEMBR
SET MEMBER_STATE = 'N'
WHERE APPLY_QUAL= qual_apply
      SET_NAME = nom_ensemble
      WHOS_ON_FIRST = qui_est_enpremier
      SOURCE_OWNER = propriétaire_source
      SOURCE_TABLE = table_source
      SOURCE_VIEW_QUAL = qual_vue_source
      TARGET_OWNER = propriétaire_cible
      TARGET_TABLE = table_cible
```

---

## Modification des propriétés d'ensembles d'abonnements

Vous pouvez modifier les propriétés d'un ensemble d'abonnements pendant que le programme Apply poursuit l'exécution et le traitement d'autres ensembles, puis réactiver l'ensemble modifié avant le cycle Apply suivant.

### A propos de cette tâche

La liste ci-après décrit les attributs que vous devrez peut-être modifier :

- Plannings d'application de mises à jour (réplication horaire ou réplication d'événements)
- Instructions d'abonnements
- Prédicats de clause WHERE pour membres d'ensembles d'abonnements
- Nombre de validations
- Valeur de blocage de données (MAX\_SYNCH\_MINUTES)

Si vous désactivez préalablement l'ensemble d'abonnements, vous évitez le traitement de l'ensemble par le programme Apply pendant que vous entrez les modifications. Le programme Apply reconnaît les modifications apportées à un ensemble d'abonnements au cours du cycle suivant la réactivation de cet ensemble.



## Procédure

Pour modifier les propriétés d'un ensemble d'abonnements, procédez comme suit :

1. Désactivez l'ensemble d'abonnements en utilisant le Centre de réplication.
2. Utilisez l'une des méthodes suivantes pour modifier l'ensemble d'abonnements :

Méthode	Description
<b>Programme de ligne de commande ASNCLP</b>	Utilisez la commande ALTER SUBSCRIPTION SET.  Les commandes suivantes définissent l'environnement et modifient l'ensemble d'abonnements SET00 pour abaisser l'intervalle de temps à 15 minutes :  SET SERVER CAPTURE TO DB SAMPLE; SET SERVER CONTROL TO DB TARGET; SET OUTPUT CAPTURE SCRIPT "capsubsetchg.sql" CONTROLSCRIPT "appsubsetchg.sql"; SET LOG "subsetchg.err"; SET RUN SCRIPT LATER; ALTER SUBSCRIPTION SET SETNAME SET00 APPLYQUAL AQ00 SETTYPE R ACTIVATE YES TIMING INTERVAL 15 COMMIT COUNT NULL;
<b>Centre de réplication</b>	Utilisez le bloc-notes Propriétés des ensembles d'abonnements. Pour l'ouvrir, cliquez sur le dossier <b>Ensembles d'abonnements</b> sur un serveur de contrôle Apply, puis cliquez avec le bouton droit de la souris sur l'ensemble d'abonnements souhaité dans le panneau de contenu ; ensuite, cliquez sur <b>Propriétés</b> .

3. Réactivez l'ensemble d'abonnements.

**z/OS** **Linux UNIX Windows** Si vous affectez au paramètre Apply **opt4one** la valeur **y**, arrêtez puis relancez le programme Apply afin que vos modifications soient reconnues.

---

## Modification des noms d'ensembles d'abonnements

Vous pouvez modifier le nom d'un ensemble d'abonnements sans pour cela devoir supprimer puis créer de nouveau l'ensemble d'abonnements et tous ses membres.

### Avant de commencer

Avant d'exécuter ces instructions SQL, familiarisez-vous avec la structure des tables de contrôle de la réplication SQL et avec les ensembles d'abonnements définis sur votre système.

**Conseil :** Si vous configurez des définitions de contrôle ou des programmes démarrés de moniteur d'alertes de réplication pour détecter des conditions d'alerte pour l'ensemble d'abonnements, supprimez ces définitions. Une fois que vous avez modifié le nom de l'ensemble d'abonnements, créez de nouveau les définitions de surveillance via le Centre de réplication ou le programme ASNCLP. Ensuite, vous pouvez réinitialiser les moniteurs à l'aide de la commande système **asnmcmd reinit**. Vous pouvez également arrêter les moniteurs à l'aide de la commande **asnmcmd stop** puis redémarrer les programmes à l'aide de la commande **asnmmon**.

## Procédure

Pour modifier le nom d'un ensemble d'abonnements, procédez comme suit :

1. Utilisez le Centre de réplication pour désactiver l'ensemble d'abonnements.
2. Sur le serveur de contrôle Apply, exécutez les instructions SQL suivantes pour modifier le nom de l'ensemble d'abonnements dans les tables IBMSNAP\_SUBS\_SET, IBMSNAP\_SUBS\_MEMBR et IBMSNAP\_SUBS\_COLS :

```

UPDATE ASN.IBMSNAP_SUBS_SET
  SET SET_NAME      = 'nouveaunomensemble'
  WHERE
    APPLY_QUAL      = 'qual_apply' AND
    SET_NAME        = 'nomensembleexistant' AND
    WHOS_ON_FIRST   = 'val';

UPDATE ASN.IBMSNAP_SUBS_MEMBR
  SET SET_NAME      = 'nouveaunomensemble'
  WHERE
    APPLY_QUAL      = 'qual_apply' AND
    SET_NAME        = 'nomensembleexistant' AND
    WHOS_ON_FIRST   = 'val';

UPDATE ASN.IBMSNAP_SUBS_COLS
  SET SET_NAME      = 'nouveaunomensemble'
  WHERE
    APPLY_QUAL      = 'qual_apply' AND
    SET_NAME        = 'nomensembleexistant' AND
    WHOS_ON_FIRST   = 'val';

```

Où *nouveaunomensemble* correspond au nouveau nom de l'ensemble d'abonnements, *qual\_apply* correspond au qualificatif Apply, *nomensembleexistant* correspond au nom existant de l'ensemble d'abonnements et *val* porte la valeur F ou S.

3. Si cet ensemble d'abonnements utilise des instructions SQL avant ou après, ou encore des appels de procédure, exécutez le script SQL suivant à partir du serveur de contrôle Apply pour modifier le nom de l'ensemble d'abonnements dans la table IBMSNAP\_SUBS\_STMTS :

```

UPDATE ASN.IBMSNAP_SUBS_STMTS
  SET SET_NAME      = 'nouveaunomensemble'
  WHERE
    APPLY_QUAL      = 'qual_apply' AND
    SET_NAME        = 'nomensembleexistant' AND
    WHOS_ON_FIRST   = 'val';

```

Où *nouveaunomensemble* correspond au nouveau nom de l'ensemble d'abonnements, *qual\_apply* correspond au qualificatif Apply, *nomensembleexistant* correspond au nom existant de l'ensemble d'abonnements et *val* porte la valeur F ou S.

4. Sur le serveur de contrôle Capture, exécutez les instructions SQL suivantes pour modifier le nom de l'ensemble d'abonnements dans les tables IBMSNAP\_PRUNE\_SET et IBMSNAP\_PRUNCNTL :

```

UPDATE Schéma.IBMSNAP_PRUNE_SET
  SET SET_NAME      = 'nouveaunomensemble'
  WHERE
    APPLY_QUAL      = 'qual_apply' AND
    SET_NAME        = 'nomensembleexistant' AND
    TARGET_SERVER   = 'serveur_cible';

UPDATE Schéma.IBMSNAP_PRUNCNTL
  SET SET_NAME      = 'nouveaunomensemble'
  WHERE
    APPLY_QUAL      = 'qual_apply' AND
    SET_NAME        = 'nomensembleexistant' AND
    TARGET_SERVER   = 'serveur_cible';

```

Où *Schéma* correspond au nom du schéma Capture, *nouveaunomensemble* correspond au nouveau nom de l'ensemble d'abonnements, *qual\_apply* correspond au qualificatif Apply, *nomensembleexistant* correspond au nom existant de l'ensemble d'abonnements et *serveur\_cible* correspond à l'emplacement de base de données des tables cible.

5. Si vous utilisez le programme Apply sous Linux, UNIX, Windows ou z/OS, affectez à **opt4one** la valeur *y*, puis arrêtez et redémarrez le programme Apply.
6. Réactivez l'ensemble d'abonnements à partir du Centre de réplication.

---

## Division d'un ensemble d'abonnements

Vous pouvez diviser un ensemble d'abonnements en deux ensembles ou plus sans devoir supprimer ou générer à nouveau les informations de cet ensemble.

### Avant de commencer

- Avant d'exécuter ces instructions SQL, familiarisez-vous avec la structure des tables de contrôle de la réplication SQL et avec les ensembles d'abonnements définis sur votre système.
- Identifiez les membres de l'ensemble d'abonnements à diviser et déterminez les tables source et cible associées à ces membres.
- Identifiez le serveur de contrôle de Capture, le serveur cible et le serveur de contrôle Apply de l'ensemble d'abonnements à diviser. Vous devez utiliser ces emplacements de serveurs (serveur de contrôle de Capture, serveur cible et le serveur de contrôle Apply) pour le nouvel ensemble d'abonnements que vous voulez créer via cette procédure.

### A propos de cette tâche

**Conseil :** Si vous configurez des définitions de contrôle ou des programmes démarrés de moniteur d'alertes de réplication pour détecter des conditions d'alerte pour l'ensemble d'abonnements, supprimez ces définitions. Une fois l'ensemble d'abonnements divisé, créez à nouveau les définitions de contrôle via le Centre de réplication ou ASNCLP. Ensuite, vous pouvez réinitialiser les moniteurs à l'aide de la commande système **asnmcmd reinit**. Vous pouvez également arrêter les moniteurs à l'aide de la commande **asnmcmd stop** puis redémarrer les programmes à l'aide de la commande **asnmon**.

### Procédure

Pour diviser un ensemble d'abonnements :

1. Désactivez-le depuis le Centre de réplication. Dans le dossier Ensembles d'abonnements, cliquez avec le bouton droit sur l'ensemble d'abonnements actif dans le panneau du contenu, puis sélectionnez Désactiver.
2. Créez un ensemble d'abonnements. Le nouvel ensemble est représenté par une nouvelle ligne dans la table IBMSNAP\_SUBS\_SET. Laissez-le inactif.
3. Depuis le serveur de contrôle Apply, exécutez l'instruction SQL suivante afin de copier des informations d'un ensemble d'abonnements existant à la nouvelle ligne dans la table IBMSNAP\_SUBS\_SET :

```
UPDATE ASN.IBMSNAP_SUBS_SET
SET STATUS =
(SELECT STATUS FROM ASN.IBMSNAP_SUBS_SET B
WHERE APPLY_QUAL = 'qual_apply' AND
SET_NAME = 'nom_existant' AND
WHOS_ON_FIRST = 'val'),
LASTRUN =
```

```

        (SELECT LASTRUN FROM ASN.IBMSNAP_SUBS_SET B
         WHERE APPLY_QUAL = 'qual_apply' AND
              SET_NAME    = 'nom_existant'    AND
              WHOS_ON_FIRST = 'val'),
SYNCHPOINT =
        (SELECT SYNCHPOINT FROM ASN.IBMSNAP_SUBS_SET B
         WHERE APPLY_QUAL = 'qual_apply' AND
              SET_NAME    = 'nom_existant'    AND
              WHOS_ON_FIRST = 'val'),
SYNCHTIME =
        (SELECT SYNCHTIME FROM ASN.IBMSNAP_SUBS_SET B
         WHERE APPLY_QUAL = 'qual_apply' AND
              SET_NAME    = 'nom_existant'    AND
              WHOS_ON_FIRST = 'val'),
LASTSUCCESS =
        (SELECT LASTSUCCESS FROM ASN.IBMSNAP_SUBS_SET B
         WHERE APPLY_QUAL = 'qual_apply' AND
              SET_NAME    = 'nom_existant'    AND
              WHOS_ON_FIRST = 'val')
WHERE
        APPLY_QUAL = 'qual_apply' AND
        SET_NAME   = 'nouveaunom' AND
        WHOS_ON_FIRST = 'val';

```

où *qual\_apply* est le qualificatif Apply, *nom\_existant* le nom de l'ensemble d'abonnements existant divisé, *val* qui a la valeur F ou S et *nouveaunom* le nom de l'ensemble d'abonnements créé.

4. Depuis le serveur de contrôle de Capture, exécutez l'instruction SQL suivante afin d'insérer une nouvelle ligne pour le nouvel ensemble d'abonnements dans la table IBMSNAP\_PRUNE\_SET :

```

INSERT INTO Schéma.IBMSNAP_PRUNE_SET
        (APPLY_QUALIFIER,
         SET_NAME,
         TARGET_SERVER,
         SYNCHTIME,
         SYNCHPOINT
VALUES ('qual_apply',
        'nouveaunom',
        'serveur_cible',
        NULL,
        x'00000000000000000000');

```

où *Schéma* est le nom du schéma de Capture, *qual\_apply* le qualificatif Apply, *nouveaunom* le nom du nouvel ensemble d'abonnements créé et *serveur\_cible* l'emplacement de la base de données des tables cible.

5. Depuis le serveur de contrôle de Capture, exécutez l'instruction SQL suivante afin de copier des informations de la ligne de l'ensemble d'abonnements existant à celle du nouvel ensemble dans la table IBMSNAP\_PRUNE\_SET :

```

UPDATE Schéma.IBMSNAP_PRUNE_SET
        SET SYNCHPOINT =
        (SELECT SYNCHPOINT FROM Schéma.IBMSNAP_PRUNE_SET B
         WHERE APPLY_QUAL = 'qual_apply' AND
              SET_NAME    = 'nom_existant'    AND
              TARGET_SERVER = 'serveur_cible'),
        SYNCHTIME =
        (SELECT SYNCHTIME FROM Schéma.IBMSNAP_PRUNE_SET B
         WHERE APPLY_QUAL = 'qual_apply' AND
              SET_NAME    = 'nom_existant'    AND
              TARGET_SERVER = 'serveur_cible')
WHERE
        APPLY_QUAL = 'qual_apply' AND
        SET_NAME   = 'nouveaunom' AND
        TARGET_SERVER = 'serveur_cible';

```

où *Schéma* est le nom du schéma de Capture, *qual\_apply* le qualificatif Apply, *nomexistant* le nom de l'ensemble d'abonnements existant divisé, *serveur\_cible* l'emplacement de la base de données des tables cible et *nouveaunom* le nom du nouvel ensemble d'abonnements créé.

6. Depuis le serveur de contrôle Apply, exécutez les instructions SQL suivantes afin de changer le nom de l'ensemble d'abonnements dans la table IBMSNAP\_SUBS\_MEMBR et les tables IBMSNAP\_SUBS\_COLS pour chaque membre déplacé vers le nouvel ensemble d'abonnements :

```
UPDATE ASN.IBMSNAP_SUBS_MEMBR
  SET SET_NAME      = 'nouveaunom'
  WHERE
    APPLY_QUAL      = 'qual_apply' AND
    SET_NAME        = 'nom_existant'      AND
    WHOS_ON_FIRST   = 'val'              AND
    SOURCE_OWNER    = 'Schémasource' AND
    SOURCE_TABLE    = 'Tablesourc'      AND
    SOURCE_VIEW_QUAL = 'qual_vue_source' AND
    TARGET_OWNER    = 'Schémacible' AND
    TARGET_TABLE    = 'Tablecible';

UPDATE ASN.IBMSNAP_SUBS_COLS
  SET SET_NAME      = 'nouveaunom'
  WHERE
    APPLY_QUAL      = 'qual_apply' AND
    SET_NAME        = 'nom_existant'      AND
    WHOS_ON_FIRST   = 'val'              AND
    TARGET_OWNER    = 'Schémacible'      AND
    TARGET_TABLE    = 'Tablecible';
```

où *nouveaunom* est le nom du nouvel ensemble d'abonnements créé, *qual\_apply* le qualificatif Apply, *nomexistant* le nom de l'ensemble d'abonnements existant divisé; *Val* qui a la valeur F ou S, *Schémasource* le schéma de la table source, *Tablesourc* le nom de la table source, *qual\_vue\_source* le qualificatif de la vue source pour cette table source, *Schémacible* le schéma de la table cible et *Tablecible* le nom de la table cible.

Répétez cette étape pour chaque membre de l'ensemble d'abonnements à déplacer vers le nouvel ensemble d'abonnements.

7. Si l'ensemble d'abonnements divisé utilise des instructions SQL avant ou après ou des appels de procédure, déplacez les instructions applicables vers le nouvel ensemble d'abonnements dans la table IBMSNAP\_SUBS\_STMTS :
  - a. Exécutez le script SQL suivant depuis le serveur de contrôle Apply pour déplacer les instructions :

```
UPDATE ASN.IBMSNAP_SUBS_STMTS
  SET SET_NAME      = 'nouveaunom'
  WHERE
    APPLY_QUAL      = 'qual_apply' AND
    SET_NAME        = 'nom_existant'      AND
    WHOS_ON_FIRST   = 'val'              AND
    STMT_NUMBER     in (instruction1,instruction2,..instructionn);
```

où *nouveaunom* est le nom du nouvel ensemble d'abonnements créé, *qual\_apply* le qualificatif Apply, *nomexistant* le nom de l'ensemble d'abonnements existant divisé, *val* qui a la valeur F ou S, et *instruction1*, *instruction2* et *instructionn* qui correspondent aux numéros d'instructions déplacées vers le nouvel ensemble d'abonnements.



- b. Modifiez les valeurs dans la colonne AUX\_STMTS de la table IBMSNAP\_SUBS\_SET de façon à refléter le nouveau nombre d'instructions pour les deux ensembles d'abonnements. Si besoin est, renumérotez les instructions pour éliminer celles en double.

8. Depuis le serveur de contrôle de Capture, exécutez l'instruction SQL suivante afin de changer le nom de l'ensemble d'abonnements dans la table IBMSNAP\_PRUNCNTL pour chaque membre déplacé :

```
UPDATE Schéma.IBMSNAP_PRUNCNTL
  SET SET_NAME      = 'nouveaunom'
 WHERE
   APPLY_QUAL      = 'qual_apply' AND
   SET_NAME        = 'nom_existant' AND
   TARGET_SERVER   = 'serveur_cible' AND
   SOURCE_OWNER    = 'Schémasource' AND
   SOURCE_TABLE    = 'Tablesource' AND
   SOURCE_VIEW_QUAL = 'qual_vue_source' AND
   TARGET_OWNER    = 'Schémacible' AND
   TARGET_TABLE    = 'Tablecible';
```

où *Schéma* est le nom du schéma de Capture, *nouveaunom* le nom du nouvel ensemble d'abonnements créé à l'étape 2, *qual\_apply* le qualificatif Apply, *nom\_existant* le nom de l'ensemble d'abonnements existant divisé, *serveur\_cible* l'emplacement de la base de données des tables cible, *Schémasource* le schéma de la table source, *Tablesourc*e le nom de la table source, *qual\_vue\_source* le qualificatif de la vue source pour cette table source de réplication, *Schémacible* le schéma de la table cible et *Tablecible* le nom de la table cible.

Répétez cette étape pour chaque membre de l'ensemble d'abonnements que vous avez déplacé vers le nouvel ensemble d'abonnements.

9.   Si vous exécutez le programme Apply et que le paramètre **opt4one** porte la valeur y, arrêtez puis redémarrez le programme Apply.
10. Réactivez les deux ensembles d'abonnements depuis le Centre de réplication.

---

## Fusion d'ensembles d'abonnements

Vous pouvez fusionner deux ensembles d'abonnements en un seul. Cette opération peut être utile si vous voulez que les tables cible dans ces ensembles présentent la même cohérence de transaction, sans pour autant devoir supprimer puis générer à nouveau les informations des ensembles.

### Avant de commencer

Avant d'exécuter ces instructions SQL, familiarisez-vous avec la structure des tables de contrôle de la réplication SQL et avec les ensembles d'abonnements définis sur votre système.

Identifiez le serveur de contrôle de Capture, le serveur cible et le serveur de contrôle Apply de chaque ensemble d'abonnements à fusionner. Vérifiez que tous les ensembles d'abonnements à fusionner ont été créés avec le même serveur de contrôle de Capture, le même serveur cible et le même serveur de contrôle Apply.

### Restrictions

Les deux ensembles d'abonnements à fusionner doivent dériver leurs données source du même serveur Capture et du même schéma de Capture.

**Important :** Ces ensembles doivent aussi avoir traité les données source jusqu'à la valeur de point de synchronisation identique pour éviter la perte de données au moment de la fusion.

## Procédure

Pour fusionner des ensembles d'abonnements :

1. Arrêtez le programme Capture associé. Patientez jusqu'à ce que deux ensembles d'abonnements atteignent le même point et le même temps de synchronisation, comme indiqué dans la table IBMSNAP\_SUBS\_SET.

**Conseil :** Pour ne pas arrêter le programme Capture, insérez un signal USER dans la table IBMSNAP\_SIGNAL et générez un événement avec END\_SYNCHPOINT (dans la table IBMSNAP\_SUBS\_EVENT) défini à la valeur de la colonne SIGNAL\_LSN dans la table IBMSNAP\_SIGNAL. De cette façon, seules les données jusqu'à ce point final sont appliquées.

2. Désactivez les deux ensembles d'abonnements du Centre de réplication.
3. Depuis le serveur de contrôle Apply, exécutez l'instruction SQL suivante pour supprimer la ligne de la table IBMSNAP\_SUBS\_SET correspondant à l'ensemble d'abonnements que vous passez à l'autre :

```
DELETE FROM ASN.IBMSNAP_SUBS_SET
WHERE
    APPLY_QUAL    = 'qual_apply' AND
    SET_NAME      = 'sous-ensemble_à_déplacer' AND
    WHOS_ON_FIRST = 'val';
```

où *qual\_apply* est le qualificatif Apply, *sous-ensemble\_à\_déplacer* le nom de l'ensemble d'abonnements que vous passez à un autre ensemble existant et *val* qui a la valeur F ou S.

4. Depuis le serveur de contrôle Capture, exécutez l'instruction SQL suivante pour supprimer la ligne de la table IBMSNAP\_PRUNE\_SET correspondant à l'ensemble d'abonnements que vous passez à l'autre :

```
DELETE FROM Schéma.IBMSNAP_PRUNE_SET
WHERE
    APPLY_QUAL    = 'qual_apply' AND
    SET_NAME      = 'sous-ensemble_à_déplacer' AND
    TARGET_SERVER = 'serveur_cible';
```

où *Schéma* est le nom du schéma de Capture, *qual\_apply* le qualificatif Apply, *sous-ensemble\_à\_déplacer* le nom de l'ensemble d'abonnements que vous passez à un autre ensemble existant et *serveur\_cible* l'emplacement de la base de données des tables cible.

5. Depuis le serveur de contrôle Apply, exécutez les instructions SQL suivantes pour renommer l'ensemble d'abonnements que vous déplacez comme l'autre ensemble dans les tables IBMSNAP\_SUBS\_MEMBR et IBMSNAP\_SUBS\_COLS :

```
UPDATE ASN.IBMSNAP_SUBS_MEMBR
SET SET_NAME      = 'sous-ensemble_fusionné_existant'
WHERE
    APPLY_QUAL    = 'qual_apply' AND
    SET_NAME      = 'sous-ensemble_à_déplacer' AND
    WHOS_ON_FIRST = 'val';

UPDATE ASN.IBMSNAP_SUBS_COLS
SET SET_NAME      = 'sous-ensemble_fusionné_existant'
WHERE
    APPLY_QUAL    = 'qual_apply' AND
    SET_NAME      = 'sous-ensemble_à_déplacer' AND
    WHOS_ON_FIRST = 'val';
```

où *sous-ensemble\_fusionné\_existant* est le nom de l'ensemble d'abonnements existant fusionné avec celui que vous déplacez, *qual\_apply* le qualificatif Apply,

*sous-ensemble\_à\_déplacer* le nom de l'ensemble d'abonnements que vous passez à l'ensemble existant et *val* qui a la valeur F ou S.

6. Si l'ensemble d'abonnements déplacé utilise des instructions SQL ou des appels de procédure antérieurs ou postérieurs, renommez-le dans la table IBMSNAP\_SUBS\_STMTS :

- a. Exécutez le script SQL suivant depuis le serveur de contrôle Apply pour renommer l'ensemble d'abonnements :

```
UPDATE ASN.IBMSNAP_SUBS_STMTS
  SET SET_NAME      = 'sous-ensemble_fusionné_existant'
 WHERE
   APPLY_QUAL      = 'qual_apply' AND
   SET_NAME        = 'sous-ensemble_à_déplacer' AND
   WHOS_ON_FIRST  = 'val';
```


où *sous-ensemble\_fusionné\_existant* est le nom de l'ensemble d'abonnements existant fusionné avec celui que vous déplacez, *qual\_apply* le qualificatif Apply, *sous-ensemble\_à\_déplacer* le nom de l'ensemble d'abonnements que vous passez à un autre ensemble existant et *val* qui a la valeur F ou S.

- b. Modifiez la valeur de la colonne AUX\_STMTS dans la table IBMSNAP\_SUBS\_SET pour refléter le nouveau nombre d'instructions dans l'ensemble d'abonnements fusionné existant. Si besoin est, renumérotez les instructions pour éliminer celles en double.

7. Depuis le serveur de contrôle de Capture, exécutez les instructions SQL suivantes pour renommer l'ensemble d'abonnements déplacé comme celui fusionné dans la table IBMSNAP\_PRUNCNTL :

```
UPDATE Schéma.IBMSNAP_PRUNCNTL
  SET SET_NAME      = 'sous-ensemble_fusionné_existant'
 WHERE
   APPLY_QUAL      = 'qual_apply' AND
   SET_NAME        = 'sous-ensemble_à_déplacer' AND
   TARGET_SERVER   = 'serveur_cible';
```

où *Schéma* est le nom du schéma de Capture, *sous-ensemble\_fusionné\_existant* le nom de l'ensemble d'abonnements existant fusionné avec celui que vous déplacez, *qual\_apply* le qualificatif Apply, *sous-ensemble\_à\_déplacer* le nom de l'ensemble d'abonnements que vous passez à un autre ensemble existant et *serveur\_cible* l'emplacement de la base de données des tables cible.

8.  Si vous exécutez le programme Apply et que le paramètre **opt4one** porte la valeur *y*, arrêtez puis redémarrez le programme Apply.
9. Réactivez l'ensemble d'abonnements fusionné depuis le Centre de réplication.

---

## Modification des qualificatifs Apply des ensembles d'abonnements

Si vous devez modifier le qualificatif Apply d'un ensemble d'abonnements, vous pouvez utiliser SQL pour apporter cette modification sans devoir supprimer puis créer de nouveau l'ensemble d'abonnements.

### Avant de commencer

Avant d'exécuter ces instructions SQL, familiarisez-vous avec la structure des tables de contrôle de la réplication SQL et avec les ensembles d'abonnements définis sur votre système.

Vous devez également déterminer les informations suivantes :



- Nom du qualificatif Apply.
- Ensembles d'abonnements à déplacer du qualificatif Apply existant vers le nouveau qualificatif Apply.
- Instructions SQL avant ou après, ou appels de procédure définis pour ces ensembles d'abonnements.

### A propos de cette tâche

Si plusieurs ensembles d'abonnements utilisent le même qualificatif Apply, il est conseillé de déplacer certains de ces ensembles vers un nouveau qualificatif Apply afin d'équilibrer la charge de travail des différents programmes Apply.

**Conseil :** Si vous avez créé des définitions de surveillance ou démarré des programmes du moniteur d'alertes de réplication en vue de détecter les critères d'alerte applicables au qualificatif Apply, supprimez ces définitions. Une fois que vous avez modifié le qualificatif, créez de nouveau les définitions de surveillance via le Centre de réplication ou le programme ASNCLP. Ensuite, vous pouvez réinitialiser les moniteurs à l'aide de la commande système **asnmcmd reinit**. Vous pouvez également arrêter les moniteurs à l'aide de la commande **asnmcmd stop** puis redémarrer les programmes à l'aide de la commande **asnmon**.

Vous devez exécuter les instructions SQL figurant dans cette procédure pour chaque ensemble d'abonnements à déplacer.

### Procédure

Pour modifier les qualificatifs Apply d'ensembles d'abonnements, procédez comme suit :

1. Désactivez les ensembles d'abonnements à modifier à l'aide du Centre de réplication.
2. Sur le serveur de contrôle Apply, exécutez les instructions SQL suivantes pour modifier le qualificatif Apply de l'ensemble d'abonnements dans les tables IBMSNAP\_SUBS\_SET, IBMSNAP\_SUBS\_MEMBR et IBMSNAP\_SUBS\_COLS :

```
UPDATE ASN.IBMSNAP_SUBS_SET
  SET APPLY_QUAL = 'nouveauqual_apply'
WHERE
  APPLY_QUAL = 'qual_applyexistant' AND
  SET_NAME = 'nom' AND
  WHOS_ON_FIRST = 'val';

UPDATE ASN.IBMSNAP_SUBS_MEMBR
  SET APPLY_QUAL = 'nouveauqual_apply'
WHERE
  APPLY_QUAL = 'qual_applyexistant' AND
  SET_NAME = 'nom' AND
  WHOS_ON_FIRST = 'val';

UPDATE ASN.IBMSNAP_SUBS_COLS
  SET APPLY_QUAL = 'nouveauqual_apply'
WHERE
  APPLY_QUAL = 'qual_applyexistant' AND
  SET_NAME = 'nom' AND
  WHOS_ON_FIRST = 'val';
```

Où *nouveauqual\_apply* correspond au nouveau qualificatif Apply, *qual\_applyexistant* correspond au qualificatif Apply existant, *nom* correspond au nom de l'ensemble d'abonnements et *val* porte la valeur F ou S.

3. Si cet ensemble d'abonnements utilise des instructions SQL avant ou après, ou encore des appels de procédure, exécutez les instructions SQL suivantes à partir

du serveur de contrôle Apply pour modifier le qualificatif Apply de l'ensemble d'abonnements dans la table IBMSNAP\_SUBS\_STMTS :

```
UPDATE ASN.IBMSNAP_SUBS_STMTS
  SET APPLY_QUAL = 'nouveauqual_apply'
WHERE
  APPLY_QUAL = 'qual_applyexistant' AND
  SET_NAME = 'nom' AND
  WHOS_ON_FIRST = 'val';
```

Où *nouveauqual\_apply* correspond au nouveau qualificatif Apply, *qual\_applyexistant* correspond au qualificatif Apply existant, *nom* correspond au nom de l'ensemble d'abonnements et *val* porte la valeur F ou S.

4. Sur le serveur de contrôle Capture, exécutez les instructions SQL suivantes pour modifier le qualificatif Apply de l'ensemble d'abonnements dans les tables IBMSNAP\_PRUNE\_SET et IBMSNAP\_PRUNCNTL :

```
UPDATE Schéma.IBMSNAP_PRUNE_SET
  SET APPLY_QUAL = 'nouveauqual_apply'
WHERE
  APPLY_QUAL = 'qual_applyexistant' AND
  SET_NAME = 'nom' AND
  TARGET_SERVER = 'serveur_cible';

UPDATE Schéma.IBMSNAP_PRUNCNTL
  SET APPLY_QUAL = 'nouveauqual_apply'
WHERE
  APPLY_QUAL = 'qual_applyexistant' AND
  SET_NAME = 'nom' AND
  TARGET_SERVER = 'serveur_cible';
```

Où *Schéma* correspond au nom du schéma Capture, *nouveauqual\_apply* correspond au nouveau qualificatif Apply, *qual\_applyexistant* correspond au qualificatif Apply existant, *nom* correspond au nom de l'ensemble d'abonnements et *serveur\_cible* correspond à l'emplacement de base de données des tables cible.

5. Répétez les étapes 2 à 4 pour chaque autre ensemble d'abonnements à déplacer.
6. Si vous utilisez le programme Apply et que le paramètre **opt4one** porte la valeur y sous Linux, UNIX, Windows ou z/OS, arrêtez puis redémarrez le programme Apply.
7. Réactivez les ensembles d'abonnements à l'aide du Centre de réplication.

---

## Désactivation d'ensembles d'abonnements

Vous pouvez désactiver un ensemble d'abonnements sans pour autant le supprimer. Lorsque vous désactivez un ensemble d'abonnements, le programme Apply termine son cycle en cours, puis suspend les opérations pour cet ensemble d'abonnements.

### Avant de commencer

Avant d'exécuter ces instructions SQL, familiarisez-vous avec la structure des tables de contrôle de la réplication SQL et avec les ensembles d'abonnements définis sur votre système.

### A propos de cette tâche

Vous devrez peut-être exécuter une maintenance spécifique sur ces ensembles d'abonnements désactivés, en fonction de la durée de désactivation prévue :

### Période courte

Il n'y a pas d'exigences spécifiques applicables aux ensembles d'abonnements temporairement désactivés. Vous devez temporairement désactiver un ensemble d'abonnements lorsque vous modifiez ses attributs ou que vous corrigez des incidents au niveau des tables cible.

Utilisez le Centre de réplication pour désactiver, modifier puis réactiver l'ensemble d'abonnements.

### Périodes longues

Vous pouvez désactiver un ensemble d'abonnements dont vous n'avez plus besoin actuellement, mais dont vous aurez besoin ultérieurement. Toutefois, vous devez effectuer une opération supplémentaire si cet ensemble d'abonnements doit rester désactivé pendant une période suffisamment longue pour que les données modifiées se soient accumulées, et que les performances des programmes Capture et Apply s'en trouvent affectées.

Le programme Capture utilise les informations des programmes Apply actifs pendant le processus d'élagage. Si les programmes Apply sont inactifs ou que les ensembles d'abonnements sont désactivés pendant de longues périodes, les informations d'élagage deviennent périmées ; en outre, les tables d'unités de travail (UOW), et éventuellement les tables de modification des données (CD), ne peuvent pas être élaguées rapidement et efficacement s'il reste des enregistrements actifs associés aux ensembles d'abonnements désactivés. Ces informations périmées risquent d'altérer gravement les performances des autres programmes Apply actifs et d'entraîner une consommation d'UC inutile et coûteuse pour l'exécution du processus d'élagage. Les tables d'unités de travail (UOW) et de modification des données (CD) sont élaguées, le cas échéant, conformément à la durée de conservation (qui a par défaut la valeur 7 jours) définie pour le programme Capture. Mais de grandes quantités de données peuvent s'accumuler au cours de cette période, en fonction de la taille de votre environnement de réplication.

Pour éviter ces incidents d'élagage, vous pouvez utiliser SQL pour réinitialiser les informations d'élagage relatives à un ensemble d'abonnements qui doit rester désactivé pendant une longue période.

Si vous avez désactivé tous les ensembles d'abonnements associés à un objet enregistré, vous devez également désactiver l'objet enregistré pour empêcher le programme Capture d'effectuer inutilement une capture de données.

### Procédure

1. Utilisez le Centre de réplication pour désactiver l'ensemble d'abonnements. Cliquez sur le dossier **Ensembles d'abonnements**, cliquez droit sur l'ensemble d'abonnements actif dans le panneau du contenu et sélectionnez **Désactiver**.
2. Sur le serveur de contrôle Capture, exécutez les instructions SQL suivantes pour réinitialiser les informations d'élagage dans les tables IBMSNAP\_PRUNE\_SET et IBMSNAP\_PRUNCNTL pour l'ensemble d'abonnements désactivé :

```
UPDATE Schéma.IBMSNAP_PRUNE_SET
  SET SYNCHPOINT = x'00000000000000000000' AND
      SYNCHTIME = NULL
WHERE
  APPLY_QUAL = 'qual_apply' AND
  SET_NAME = 'nom' AND
  TARGET_SERVER = 'serveur_cible';
```

```

UPDATE Schéma.IBMSNAP PRUNCNTL
   SET SYNCHPOINT = NULL AND
       SYNCHTIME  = NULL
WHERE
   APPLY_QUAL    = 'qual_apply' AND
   SET_NAME      = 'nom'          AND
   TARGET_SERVER = 'serveur_cible';

```

Où *Schéma* correspond au nom du schéma Capture, *qual\_apply* correspond au qualificatif Apply, *nom* correspond au nom de l'ensemble d'abonnements et *serveur\_cible* correspond à l'emplacement de base de données des tables cible.

## Suppression d'ensembles d'abonnements

Si vous n'avez plus besoin de répliquer les données dans un ensemble d'abonnements déterminé, vous pouvez supprimer ce dernier. Toutefois, si votre programme Apply traite l'ensemble d'abonnements que vous supprimez, son travail s'interrompt et tous les autres ensembles d'abonnements dans celui-ci ne sont pas traités tant que vous ne relancez pas ce travail.

### Procédure

Pour supprimer des ensembles d'abonnements :

1. Pour vérifier que le programme Apply a terminé tout traitement en cours pour l'ensemble d'abonnements, désactivez ce dernier avant de le supprimer depuis le Centre de réplication. Cliquez sur le dossier **Ensembles d'abonnements**, cliquez droit sur l'ensemble d'abonnements actif dans le panneau du contenu et sélectionnez **Désactiver**.
2. Utilisez l'une des méthodes suivantes pour supprimer un ensemble d'abonnements désactivé :

Méthode	Description
<b>Programme de ligne de commande ASNCLP</b>	Utilisez la commande DROP SUBSCRIPTION SET.  Les commandes suivantes définissent l'environnement et supprime un ensemble d'abonnements SET00 avec le qualificatif Apply AQ00.  SET SERVER CAPTURE TO DB SAMPLE; SET SERVER CONTROL TO DB TARGET; SET OUTPUT CAPTURE SCRIPT "drpcapsubset.sql" CONTROLSCRIPT "drpappsubset.sql"; SET LOG "drpsubset.err"; SET RUN SCRIPT LATER; DROP SUBSCRIPTION SET SETNAME SET00 APPLYQUAL AQ00;
<b>Centre de réplication</b>	Utilisez la fenêtre Suppression d'ensembles d'abonnements. Pour ouvrir la fenêtre, cliquez sur le dossier <b>Ensembles d'abonnements</b> , cliquez avec le bouton droit sur l'ensemble d'abonnements actif dans le panneau du contenu et sélectionnez <b>Supprimer</b> .
<b>System i</b>  Commande système <b>RMVDPRSUB</b>	Utilisez la commande Remove DPR Subscription Set (RMVDPRSUB) pour supprimer un ensemble d'abonnements.

Le programme Capture poursuit la capture de données et l'écriture de lignes dans la table CD même si vous supprimez tous les ensembles d'abonnements pour

l'objet enregistré. Pour empêcher ce traitement continu par le programme Capture, désactivez ou supprimez l'objet enregistré après avoir supprimé ses ensembles d'abonnements.

---

## Coordination entre les événements de réplication et les événements d'applications de base de données

Vous pouvez coordonner les événements de réplication et les événements de base de données en insérant manuellement des lignes dans la table `IBMSNAP_SIGNAL`. Ces lignes, appelées signaux, indiquent aux programmes Capture en cours de fonctionnement les actions spécifiques à entreprendre.

### Définition d'un événement `END_SYNCHPOINT` à l'aide du signal `USER`

Vous pouvez affecter la valeur `USER` à la colonne `SIGNAL_TYPE` afin d'établir un point précis dans le journal de reprise DB2 et de coordonner un événement de réplication avec un événement d'application de base de données.

#### A propos de cette tâche

Par exemple, si vous effectuez une réplication de données OLTP (Online Transaction Processing) dans un magasin de données géré séparément, vous pouvez maintenir une bonne stabilité des données en vue du traitement des requêtes. Vous effectuez la mise à jour des données du magasin à l'aide uniquement des modifications apportées à un moment spécifique d'une journée de travail de l'application OLTP. Dans ce cas, l'événement d'application de base de données est la fin logique de la journée de travail. L'événement de réplication représente l'application des modifications entre la clôture de l'activité d'un jour spécifique et la clôture de l'activité du jour suivant. Supposons que les ensembles d'abonnements soient configurés pour le traitement des événements uniquement.

#### Procédure

Pour créer un signal de type `USER`, procédez comme suit :

1. Créez un signal Capture de type `USER` en insérant la ligne suivante dans la table `IBMSNAP_SIGNAL` :

```
INSERT INTO Schéma.IBMSNAP_SIGNAL
      (signal_type,
       signal_subtype,
       signal_state)
VALUES('USER',
      'USER APPLY EVENT SIGNAL',
      'P');
```

Exécutez cette instruction SQL `INSERT` lorsque l'événement d'application de base de données survient (dans ce cas, à la fin de la journée de travail de l'application).

Le programme Capture agit sur cet enregistrement de journal de table de signaux une fois que le programme Capture a trouvé l'enregistrement dans le journal de récupération de base de données (uniquement si le programme Capture trouve l'enregistrement de validation correspondant à cette insertion), et prend soin de vérifier que cet événement a été validé.

Lorsqu'un signal de type `USER` est validé, le programme Capture met à jour les valeurs suivantes de la colonne `IBMSNAP_SIGNAL` correspondant à l'enregistrement d'insertion en cours de traitement :

- SIGNAL\_STATE = 'R' (reçu par le programme Capture)
  - SIGNAL\_LSN = numéro de séquence de journal issu de l'enregistrement de validation de l'unité de travail DB2 contenant cette insertion de ligne de signal
2. Utilisez la valeur qui figure désormais dans la colonne SIGNAL\_LSN de la ligne de signal insérée pour inclure une valeur END\_SYNCHPOINT dans la table de contrôle IBMSNAP\_SUBS\_EVENT. Cette nouvelle valeur indique au programme Apply que toutes les données relatives au nouveau jour de travail ont été collectées par le programme Capture et que le programme Apply doit extraire et appliquer les données jusqu'à la valeur de la colonne SIGNAL\_LSN uniquement.

Vous pouvez automatiser l'insertion dans la table IBMSNAP\_SUBS\_EVENT en créant un déclencheur de mise à jour dans la table IBMSNAP\_SIGNAL :

```
CREATE TRIGGER EVENT_TRIG
NO CASCADE AFTER UPDATE ON Schéma.IBMSNAP_SIGNAL
REFERENCING NEW AS N
FOR EACH ROW MODE DB2SQL
WHEN (N.SIGNAL_SUBTYPE = 'USER APPLY EVENT SIGNAL')
INSERT INTO ASN.IBMSNAP_SUBS_EVENT VALUES
('WH APPLY EVENT',
(CURRENT_TIMESTAMP + 2 MINUTES),
N.SIGNAL_LSN,
null);
```

Ce déclencheur est activé chaque fois que la table IBMSNAP\_SIGNAL est mise à jour par le programme Capture. Lorsqu'une colonne SIGNAL\_SUBTYPE est mise à jour (USER APPLY EVENT SIGNAL), le déclencheur insère une ligne dans la table IBMSNAP\_SUBS\_EVENT. Cette ligne indique au programme Apply qu'il doit rechercher et appliquer le travail de la dernière journée d'activité (qui a été validé par le programme Capture) après un délai de deux minutes.

## Utilisation du signal Capture CMD STOP

Vous pouvez affecter la valeur CMD à la colonne SIGNAL\_TYPE et la valeur STOP à la colonne SIGNAL\_SUBTYPE afin d'arrêter un processus du programme Capture à un point précis dans le journal de récupération DB2.

Vous pouvez utiliser le signal Capture CMD STOP dans les situations suivantes :

- Pour obtenir une coordination entre le programme Capture et les modifications de table source, qui rendent les enregistrements de journal précédents illisibles. Cela peut se produire en cas de suppression puis de nouvelle création d'une table, ou encore de réorganisation de table sans affectation de la valeur YES à l'option KEEPDICTIONARY.
- Pour définir un point de récupération commun entre différents systèmes de bases de données réparties répliquées.

### Coordination d'une modification de table source avec le programme Capture

Vous pouvez utiliser un signal Capture de type CMD et sous-type STOP pour arrêter un programme Capture et pour coordonner les modifications de table source.

#### Procédure

Pour coordonner des modifications de tables source, procédez comme suit :

1. Créez un signal Capture de type CMD et sous-type STOP ; pour cela, insérez une ligne dans la table IBMSNAP\_SIGNAL à l'aide de l'instruction SQL suivante :

```
INSERT INTO Schéma.IBMSNAP_SIGNAL
      (signal_type,
       signal_subtype,
       signal_state)
VALUES('CMD',
       'STOP',
       'P');
```

Vous devez insérer cette ligne lorsque l'événement d'application de base de données se produit, après la mise au repos de l'activité de la table source mais avant l'activité entraînant des modifications d'enregistrements de journal problématiques.

Le programme Capture agit sur cet enregistrement de journal de table de signaux une fois que le programme Capture a trouvé l'enregistrement dans le journal de récupération de base de données (uniquement si le programme Capture trouve l'enregistrement de validation correspondant à cette insertion), et prend soin de vérifier que cet événement a été validé.

Le programme Capture arrête toutes les unités d'exécution Capture dans l'ordre, après avoir validé toutes les données capturées dans les transactions (antérieures à l'enregistrement de validation de l'unité de travail DB2 contenant cette ligne IBMSNAP\_SIGNAL insérée). Avant de s'arrêter, le programme Capture met également à jour les valeurs suivantes sur la ligne de table IBMSNAP\_SIGNAL correspondant à l'enregistrement d'insertion en cours de traitement :

- SIGNAL\_STATE = 'R' (reçu par le programme Capture)
- SIGNAL\_LSN = numéro de séquence de journal issu de l'enregistrement de validation de l'unité de travail DB2 contenant cette insertion de ligne de signal

Tous les enregistrements de journal concernant la table source modifiée sont traités par le programme Capture lorsque celui-ci prend fin.

2. Selon votre scénario, vous pouvez supprimer puis créer de nouveau votre table source, ou encore la réorganiser et la compresser sans affecter la valeur YES à l'option KEEPDICTIONARY.
3. Si vous avez supprimé ou modifié certaines colonnes répliquées, vous devez maintenant modifier les ensembles correspondants d'enregistrements et d'abonnements créés pour cette table source. En cas de besoin, ces modifications peuvent être coordonnées avec le programme Apply : dans ce cas, il convient d'attendre que les ensembles d'abonnements affectés intègrent les modifications dans le programme Capture (à l'arrêt). Un ensemble d'abonnements est synchronisé avec le programme Capture lorsque la valeur de la colonne SYNCHPOINT de la table IBMSNAP\_SUBS\_SET est égale à la valeur de la colonne MAX\_COMMITSEQ de la table Schéma.IBMSNAP\_RESTART.

### Définition d'un point de reprise distribué

Vous pouvez utiliser un signal Capture de type CMD et sous-type STOP pour définir les bases de données source et cible avec des points de reprise équivalents et pour effectuer la reprise des bases de données à un point commun de cohérence.

#### Avant de commencer

Avant d'utiliser cette procédure, vérifiez que vos tables de contrôle Apply ont été créées dans la base de données cible.

Vérifiez également que toutes les activités relatives à la base de données source ont été mises au repos avant d'insérer la ligne dans la table IBMSNAP\_SIGNAL. Cependant, ne créez pas la copie de sauvegarde des tables de base de données avant d'avoir inséré la ligne dans la table IBMSNAP\_SIGNAL.

Si vos ensembles d'abonnements ne sont pas configurés pour le traitement d'événements, vous devez temporairement les définir en conséquence. Utilisez l'instruction SQL suivante pour insérer une ligne dans la table d'événements d'abonnements IBMSNAP\_SUBS\_EVENT :

```
INSERT
INTO ASN.IBMSNAP_SUBS_EVENT
VALUES('RECOVERY_EVENT',
CURRENT_TIMESTAMP + 2 MINUTES,
valeur_SIGNAL_LSN,
NULL);
```

Où *valeur\_SIGNAL\_LSN* correspond au numéro LSN défini par le programme Capture et enregistré dans la table IBMSNAP\_SIGNAL.

### Procédure

Pour définir un point de reprise distribué, procédez comme suit :

1. Créez un signal Capture de type CMD et sous-type STOP ; pour cela, insérez une ligne dans la table IBMSNAP\_SIGNAL à l'aide de l'instruction SQL suivante :

```
INSERT INTO Schéma.IBMSNAP_SIGNAL
(signal_type,
signal_subtype,
signal_state)
VALUES('CMD',
'STOP',
'P');
```

Le programme Capture agit sur cet enregistrement de journal de table de signaux une fois que le programme Capture a trouvé l'enregistrement dans le journal de récupération de base de données (uniquement si le programme Capture trouve l'enregistrement de validation correspondant à cette insertion), et prend soin de vérifier que cet événement a été validé.

Le programme Capture arrête toutes les unités d'exécution Capture dans l'ordre, après avoir validé toutes les données capturées dans les transactions (antérieures à l'enregistrement de validation de l'unité de travail DB2 contenant cette ligne IBMSNAP\_SIGNAL insérée). Avant de s'arrêter, le programme Capture met également à jour les valeurs suivantes sur la ligne de table IBMSNAP\_SIGNAL correspondant à l'enregistrement d'insertion en cours de traitement :

- SIGNAL\_STATE = 'R' (reçu par le programme Capture)
- SIGNAL\_LSN = numéro de séquence de journal issu de l'enregistrement de validation de l'unité de travail DB2 contenant cette insertion de ligne de signal

Tous les enregistrements de journal concernant la base de données source sont traités par le programme Capture lorsque celui-ci prend fin.

2. Exécutez la sauvegarde de la base de données source ou les utilitaires de copie d'image.
3. Utilisez les valeurs figurant dans la colonne SIGNAL\_LSN de la ligne de table IBMSNAP\_SIGNAL (que vous avez insérée en tant que valeur END\_SYNCHPOINT dans la table IBMSNAP\_SUBS\_EVENT). Cette valeur



indique au programme Apply que toutes les données relatives au nouveau jour de travail ont été collectées par le programme Capture et que le programme Apply doit extraire et appliquer les données jusqu'à la valeur de la colonne SIGNAL\_LSN uniquement. Les ensembles d'abonnements traitent alors toutes les données jusqu'à la valeur SIGNAL\_LSN.

4. Exécutez la sauvegarde de la base de données cible ou les utilitaires de copie d'image. Les bases de données source et cible possèdent désormais des points de reprise équivalents, et vous pouvez effectuer la reprise des deux bases de données à un point commun de cohérence.

Vous pouvez reprendre l'activité de la base de données source dès que les événements Apply ont été définis et que la sauvegarde de la base de données source ou l'utilitaire de copie d'image prend fin. Vous pouvez également démarrer le programme Capture. Une fois terminée la sauvegarde de la base de données cible ou l'exécution de l'utilitaire de copie d'image, vous pouvez rétablir les valeurs d'origine des options de planification de vos ensembles d'abonnements (basés sur le temps, basés sur les événements, ou les deux).

**System i** Vous pouvez envoyer le signal STOP pour arrêter un travail de journal ou encore tous les travaux de journaux. Pour arrêter un travail de journal, insérez le signal dans la table des signaux correspondant à ce journal (la table IBMSNAP\_SIGNAL\_xxx\_yyy, où xxx correspond à la bibliothèque de journal et yyy correspond au nom du journal). Pour arrêter tous les travaux de journaux, insérez le signal dans la table *schéma*.IBMSNAP\_SIGNAL. Pour arrêter un travail de journal au sein d'une configuration de journaux distante, insérez le signal dans la table de signaux de journal, sur le serveur source. Examinez la description de la méthode de création de tables de signaux de journal au sein d'une configuration de journaux distante.

## Exécution d'un signal d'établissement de liaison CAPSTART hors du programme Apply

Avant que le programme Apply puisse utiliser un ensemble d'abonnements pour rechercher et appliquer des modifications des tables de modification des données (CD), un établissement de liaison (communication synchronisée) doit avoir lieu entre le programme Capture et le programme Apply, pour chaque membre d'ensemble d'abonnements concerné.

### A propos de cette tâche

Le programme Apply lance l'établissement de liaison en insérant un signal de type CMD (sous-type CAPSTART) dans la table IBMSNAP\_SIGNAL. Le programme Apply insère ce signal avant d'effectuer la régénération intégrale d'un membre d'ensemble d'abonnements dont la table cible est définie comme complète.

### Procédure

Pour effectuer l'établissement d'une liaison CAPSTART hors du programme Apply, procédez comme suit :

Créez un signal Capture de type CMD et sous-type CAPSTART ; pour cela, insérez une ligne dans la table IBMSNAP\_SIGNAL à l'aide de l'instruction SQL suivante :

```
INSERT INTO Schéma.IBMSNAP_SIGNAL
    (signal_type,
     signal_subtype,
     signal_input_in,
```

```

        signal_state)
VALUES('CMD',
      'CAPSTART',
      mapid,
      'P' );

```

Où *mapid* correspond à la valeur de colonne MAP\_ID de la table *Schéma*.IBMSNAP\_PRUNCNTL et représente la ligne du membre d'ensemble d'abonnements nécessitant l'établissement de liaison.

**Remarque :** Exécutez cette instruction SQL INSERT avant d'effectuer une régénération intégrale du membre d'ensemble d'abonnements, le cas échéant. Le programme Capture agit sur cet enregistrement de journal de table de signaux une fois que le programme Capture a trouvé l'enregistrement dans le journal de récupération de base de données (uniquement si le programme Capture trouve l'enregistrement de validation correspondant à cette insertion), et prend soin de vérifier que cet événement a été validé.

Le programme Capture vérifie si l'enregistrement correspondant a déjà été placé en mémoire sur la base de l'utilisation précédente de la table enregistrée. Si la table enregistrée n'est pas en cours d'utilisation, le programme Capture lit les informations d'enregistrement associées en mémoire et définit les valeurs de la table IBMSNAP\_REGISTER afin d'indiquer que cette table enregistrée est désormais active et en cours d'utilisation.

Que cette table enregistrée soit en cours d'utilisation ou non, le programme Capture définit les valeurs des colonnes SYNCHPOINT et SYNCHTIME sur la ligne associée de la table *Schéma*.IBMSNAP\_PRUNCNTL en spécifiant le numéro de l'enregistrement de journal de validation applicable à l'unité de travail DB2 qui contient cette ligne de signal et l'horodatage de ce même enregistrement, respectivement.

Le programme Capture met à jour les valeurs suivantes sur la ligne de table IBMSNAP\_SIGNAL correspondant à l'enregistrement de journal en cours de traitement :

- SIGNAL\_STATE = 'C' (reçu et complété par le programme Capture)
- SIGNAL\_LSN = numéro de séquence de journal issu de l'enregistrement de validation de l'unité de travail DB2 contenant cette insertion de ligne de signal

## Exécution d'un signal CAPSTOP

Vous pouvez démarrer un signal CAPSTOP si vous souhaitez arrêter manuellement la capture des modifications pour un enregistrement. Vous pouvez utiliser ce signal lorsque vous désactivez un enregistrement, ou encore avant la suppression d'un enregistrement.

### Procédure

Pour exécuter un signal CAPSTOP, procédez comme suit :

1. Créez un signal Capture de type CMD et sous-type CAPSTOP ; pour cela, insérez une ligne dans la table IBMSNAP\_SIGNAL à l'aide de l'instruction SQL suivante :

```

INSERT INTO Schéma.IBMSNAP_SIGNAL
        (signal_type,
         signal_subtype,
         signal_input_in,
         signal_state)
VALUES('CMD',
      'CAPSTOP',
      propriétaire_source.table_source,
      'P' );

```

Où *Schéma* correspond au nom du schéma Capture et *propriétaire\_source.table\_source* correspond au nom qualifié complet de la table qui n'a plus besoin des modifications capturées.

Le programme Capture agit sur cet enregistrement de journal de table de signaux une fois que le programme Capture a trouvé l'enregistrement dans le journal de récupération de base de données (uniquement si le programme Capture trouve l'enregistrement de validation correspondant à cette insertion), et prend soin de vérifier que cet événement a été validé.

Le programme Capture vérifie si l'enregistrement correspondant a déjà été placé en mémoire sur la base de l'utilisation précédente de la table enregistrée. Si la table enregistrée n'est pas en cours d'utilisation, le programme Capture ignore le signal CAPSTOP.

Si la table enregistrée est en cours d'utilisation, le programme Capture efface la mémoire associée à cet enregistrement et désactive l'enregistrement (en affectant à la colonne STATE de la table IBMSNAP\_REGISTER la valeur I). Le programme Capture arrête alors de capturer les modifications pour cette table enregistrée.

Le programme Capture met à jour les valeurs suivantes sur la ligne de table IBMSNAP\_SIGNAL correspondant à l'enregistrement de journal en cours de traitement :

- SIGNAL\_STATE = 'C' (reçu et complété par le programme Capture)
  - SIGNAL\_LSN = numéro de séquence de journal issu de l'enregistrement de validation de l'unité de travail DB2 contenant cette insertion de ligne de signal
2. Facultatif : Facultatif : supprimez l'enregistrement.
  3. **System i** Facultatif : Vous pouvez également envoyer un signal CAPSTOP pour arrêter la capture des modifications pour un enregistrement via l'insertion du signal dans la table IBMSNAP\_SIGNAL\_XXXX\_YYYY, où XXXX correspond à la bibliothèque de journal et YYYY correspond au nom de journal. Pour arrêter la capture des modifications pour un enregistrement au sein d'une configuration de journaux distante, insérez le signal CAPSTOP sur le serveur source.

---

## Réglage de l'heure d'été/d'hiver (System i)

Sur System i, le programme Capture utilise un horodatage et un numéro de séquence de journal lorsqu'il lit les modifications à partir d'un journal. Ce processus peut créer des problèmes lorsqu'il est nécessaire de régler l'horloge du système pour l'heure d'été /hiver américaine, en automne et au printemps.

### A propos de cette tâche

Les systèmes System i proposent deux méthodes pour le réglage de l'heure d'été/d'hiver :

**V5R3** Le système retarde son horloge (automne) ou l'avance (printemps) pour éviter d'ignorer ou de dupliquer des horodatages. Si vous exécutez le programme Capture sur System i V5R3 et que vous utilisez cette nouvelle méthode pour effectuer le changement d'heure, vous n'avez pas besoin d'utiliser la procédure ci-dessous.

### Avant V5R3

Vous devez arrêter toute l'activité du système pendant une heure puis retarder l'horloge d'une heure en automne. Avec cette méthode, vous devez utiliser la procédure ci-dessous.

## Procédure

Pour régler l'heure d'hiver :

1. Suivez ces étapes lorsque vous devez retarder l'horloge d'une heure en automne :
  - a. Arrêtez le programme Capture ainsi que toutes les applications qui mettent à jour les tables source.
  - b. Attendez que le temps système avance d'au moins une heure sans ajouter de nouvelles entrées de journal dans le journal source.
  - c. Remontez le temps du système d'une heure.
  - d. Redémarrez le programme Capture.

L'exemple suivant montre comment utiliser cette procédure :

- a. A 12:00 vous arrêtez le programme Capture ainsi que toutes les applications.
- b. Vous attendez jusqu'à 13:00 afin que les horodatages des entrées de journal n'aient des valeurs que jusqu'à 12:00.
- c. Vous remontez l'horloge du système à 12:00.
- d. Vous effectuez une modification. L'horodatage des entrées de journal pour la modification sera 12:01.
- e. Vous redémarrez Capture. Capture démarrera à partir de 12:00 et capturera donc les modifications intervenues à 12:01 (avant passage à l'heure d'été), ce qui correspond à 13:01 après passage à l'heure d'été.

Le programme Capture redémarre avec un horodatage antérieur au temps système en cours. Aucune entrée de journal ne sera ajoutée tant que le nouveau temps système est supérieur au temps système juste avant le changement d'heure, c'est pourquoi il n'y a aucun risque de perdre des données.

**Recommandation :** Bien que le changement d'heure n'ait pas d'effet sur le programme Apply, arrêtez et redémarrez également le programme Apply pendant le changement d'heure.

2. Suivez cette procédure lorsque vous devez avancer l'horloge d'une heure pour le passage à l'heure d'été (au printemps) :
  - a. Arrêtez le programme Capture et procédez au changement d'heure. Le programme Capture réagit comme si une heure avait passé sans changement dans les tables source.

---

## Options de promotion de votre configuration de réplication dans un autre système

Lorsque vous définissez des objets enregistrés ou des ensembles d'abonnements sur un système (système de test, par exemple) et que vous devez copier l'environnement de réplication sur un autre système (système de production, par exemple), vous pouvez utiliser les fonctions de promotion du Centre de réplication.

Les fonctions de promotion ont recours à l'ingénierie inverse pour créer des fichiers scripts en langage DDL (data definition language) et DML (data manipulation language) pour vos objets enregistrés ou vos ensembles d'abonnements. Vous pouvez copier les définitions de réplication dans une autre base de données sans avoir à enregistrer de nouveau les source ou à créer de nouveau les ensembles d'abonnements.

Par exemple, vous pouvez utiliser les fonctions de promotion pour définir des ensembles d'abonnements pour bases de données cible distantes. Une fois que vous avez défini un modèle de système cible dans votre environnement de test, vous pouvez créer des scripts d'ensembles d'abonnements (et modifier notamment le qualificatif Apply utilisé) pour vos systèmes cible distants (qui ne sont par ailleurs pas pris en charge à partir d'un point de contrôle central).

**Important :** Les fonctions de promotion ne se connectent pas au système cible et ne valident pas les paramètres de configuration de la réplication applicables à ce système.

La liste ci-après décrit les trois options de promotion de la configuration de réplication sur un autre système.

#### **Promouvoir les tables enregistrées**

Cette fonction assure la promotion des informations d'enregistrement des tables spécifiées. Elle assure également, de façon facultative, la promotion des définitions de tables de base, d'index et d'espaces table. Vous pouvez spécifier un autre schéma Capture et un autre nom de serveur pour les tables dont vous souhaitez assurer la promotion. Vous pouvez également modifier le nom de schéma des tables de modification des données (CD) associées aux tables source promues.

Vous avez la possibilité de promouvoir plusieurs tables enregistrées simultanément. Les nouveaux noms de schéma indiqués sont communiqués à toutes les tables promues.

Cette fonction effectue la promotion de tables enregistrées dans DB2 (Version 8 ou ultérieure).

#### **Promouvoir les vues enregistrées**

Cette fonction assure la promotion des informations d'enregistrement pour les vues spécifiées. Elle assure également, de façon facultative, la promotion des définitions de vues de base, de tables de base non enregistrées, d'index et d'espaces table. Vous pouvez spécifier un autre schéma Capture et un autre nom de serveur pour les vues dont vous souhaitez assurer la promotion. Vous pouvez également modifier le nom de schéma des vues de modification des données (CD) associées aux vues source promues et des tables CD que ces vues CD prennent comme base.

Vous avez la possibilité de promouvoir plusieurs vues enregistrées simultanément. Les nouveaux noms de schéma indiqués sont communiqués à toutes les vues promues.

**Important :** Si la vue dont vous assurez la promotion prend comme base une table source enregistrée, vous devez promouvoir séparément la table source enregistrée, via l'utilisation de la fonction de promotion de tables enregistrées. Ces tables source enregistrées ne sont pas promues automatiquement par la fonction de promotion des vues enregistrées. Cependant, les tables de base non enregistrées (que les vues prennent comme base) sont promues par cette fonction, en cas de besoin.

#### **Promouvoir les ensembles d'abonnements**

Cette fonction permet de promouvoir les ensembles d'abonnements. Elle vous permet de copier un ensemble d'abonnements (et tous ses membres) d'une base de données dans une autre.

Vous devez utiliser la fonction de promotion des ensembles d'abonnements avec la fonction de promotion des tables enregistrées.

**Important :** Vous pouvez utiliser les fonctions de promotion pour promouvoir les objets et les ensembles d'abonnements enregistrés résidant sous tous les systèmes d'exploitation existants. Les fonctions de promotion copient les définitions de réplication entre systèmes identiques uniquement (par exemple d'un système DB2 pour z/OS vers un autre système DB2 pour z/OS).

Vous ne pouvez pas utiliser les fonctions de promotion pour copier les définitions de répliques de ou vers une base de données relationnelle non DB2. De plus, vous ne pouvez pas non plus utiliser les fonctions de promotion pour copier des définitions de réplication qui incluent des journaux distants System i.

---

## Chapitre 13. Gestion d'un environnement de réplication SQL

Vous devez gérer les systèmes source, les tables de contrôle et les tables cibles qui résident sur votre base de données et sont utilisées par la réplication SQL.

La réplication SQL fonctionne avec votre système de base de données et n'exige que des changements mineurs de vos activités de base de données existantes. Toutefois, pour que votre système continue de fonctionner correctement et pour éviter tout incident potentiel, déterminez les besoins de votre environnement de réplication, ainsi que l'impact potentiel de ces besoins sur votre système de base de données.

Les rubriques suivantes décrivent les besoins de maintenance des systèmes source, des tables de contrôle et des tables cible.

---

### Gestion de systèmes source

Le système de réplication source contient le mécanisme de capture des modifications, les tables source à répliquer (y compris les journaux distants utilisés sur les systèmes System), les données de journal utilisées par le programme Capture, ainsi que les déclencheurs Capture utilisés au niveau des sources de base de données relationnelles non DB2.

Ces rubriques expliquent comment gérer correctement vos tables source et vos fichiers journaux et comment vous assurer que ces tables et fichiers sont toujours accessibles pour la réplication SQL.

### Accès aux tables et aux vues source

Vous devez tenir compte de la disponibilité des tables source dans les environnements de réplication SQL, afin que les programmes Capture et Apply soient toujours à même de fonctionner.

Les objets de réplication source représentent des tables et des vues de base de données qui nécessitent la même maintenance que les autres tables et vues de base de données de votre système. Poursuivez l'exécution des utilitaires et des routines de maintenance existants sur ces objets.

La réplication SQL ne requiert pas un accès direct aux tables source durant la plupart des tâches de traitement. Cependant, la réplication SQL doit accéder directement aux tables ou aux espaces table source lorsque le programme Apply effectue une actualisation complète.

### Journaux source et récepteurs de journal

Vos journaux de récupération DB2 ont deux objectifs : fournir des fonctions de récupération DB2 et fournir des informations aux programmes Capture en cours de fonctionnement.

Vous devez conserver les données du journal pour la reprise de DB2 et pour la réplication SQL ; vous devez également vous assurer que les programmes Capture et DB2 sont entièrement exécutés avec un ensemble de journaux ou de récepteurs de journal avant de procéder à leur suppression.

**Remarque :** La réplication SQL n'utilise pas les données de journal issues de bases de données relationnelles non DB2.

### **Conservation des données de journal (Linux, UNIX, Windows)**

Les données de journal résident dans les mémoires tampon de journal, les journaux actifs ou les journaux archivés. Chaque fois que le programme Capture démarre à chaud, il requiert tous les journaux DB2 créés depuis son arrêt et tous les journaux DB2 qu'il n'a pas complètement traités.

#### **Avant de commencer**

**Remarque :** Vous devez configurer votre base de données en vue de l'utilisation de l'archivage d'exit utilisateur, afin que vos programmes Capture puissent extraire des données des journaux archivés.

#### **A propos de cette tâche**

Si vous exécutez le programme Capture tandis que DB2 est en cours de fonctionnement, il est généralement en phase avec les journaux de reprise DB2. Si vous exécutez les programmes Capture lorsque DB2 est actif, ou si vous conservez les enregistrements de journal pendant une semaine ou davantage, 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 SQL 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.

#### **Procédure**

Pour déterminer les enregistrements de journal qui doivent être conservés afin d'être utilisés par le programme Capture et les enregistrements de journal qui peuvent être supprimés, procédez comme suit :

1. Exécutez l'instruction SQL suivante afin d'obtenir la valeur de MIN\_INFLIGHTSEQ contenue dans la table IBMSNAP\_RESTART :

**Pour les bases de données partitionnées :** Au sein d'un environnement composé de plusieurs partitions, cette procédure doit être étendue à chaque partition, car chaque partition gère son propre ensemble de fichiers journaux. Utilisez la colonne SEQUENCE de la table IBMSNAP\_PARTITIONINFO pour déterminer ces informations pour chaque partition.table.

```
SELECT MIN_INFLIGHTSEQ  
FROM ASN.IBMSNAP_RESTART  
WITH UR;
```

La valeur de MIN\_INFLIGHTSEQ s'affiche. Cette valeur est une colonne CHAR(10) FOR BIT DATA, composée de 20 caractères hexadécimaux. Par exemple :

00000000123456123456

Notez les 12 derniers caractères minimum de la valeur de MIN\_INFLIGHTSEQ.  
Dans l'exemple :

123456123456



**Avertissement :** Le programme Capture met à jour la valeur de IBMSNAP\_RESTART à chaque validation de données, sur la base de la valeur du paramètre **commit\_interval** . L'instruction SQL utilisée dans cette procédure spécifie une lecture non validée ; par conséquent, il est possible que vous receviez une valeur non validée pour MIN\_INFLIGHTSEQ. Pour vérifier que vous disposez de la valeur la plus précise, exécutez l'instruction SELECT, attendez la fin de l'intervalle de validation, puis exécutez de nouveau l'instruction SELECT. Utilisez la valeur la plus faible de MIN\_INFLIGHTSEQ pour le reste de cette procédure.

2. A partir d'une ligne de commande, saisissez la commande **db2 get db cfg** afin d'obtenir le chemin d'accès aux fichiers journaux actifs. Par exemple :  
`db2 get db cfg for votrenombd`

Où *votrenombd* représente le nom de votre base de données. Notez le chemin d'accès aux fichiers journaux actifs figurant dans la sortie qui s'affiche à l'écran. Par exemple :

Chemin d'accès aux fichiers journaux =C:\DB2\NODE0000\SQL00001\SQLLOGDIR\

3. A partir d'une ligne de commande DB2, saisissez la commande **db2f1sn** et entrez les 12 derniers caractères de la valeur de MIN\_INFLIGHTSEQ. Par exemple :  
`C:\DB2\NODE0000\SQL00001>db2f1sn 123456123456`

Pour exécuter la commande **db2f1sn**, vous devez avoir accès au fichier SQLLOGCTL.LFH.1 ou à sa copie miroir SQLLOGCTL.LFH.2. Les deux fichiers sont situés dans le répertoire de base de données. Le système extrait et affiche le nom du fichier contenant l'enregistrement de journal identifié par le numéro de séquence du journal. Par exemple :

Ce numéro figure dans le fichier journal S000123.LOG

## Accès aux récepteurs de journal (System i)

Il est essentiel de conserver tous les récepteurs de journal nécessaires au programme Capture.

Lorsque vous redémarrez le programme Capture avec le paramètre RESTART(\*YES), le programme Capture continue le traitement à partir de l'endroit où il s'était précédemment arrêté et requiert tous les récepteurs de journal utilisés par une ou plusieurs des tables source.

Pour vous assurer que votre programme Capture peut accéder à tous les récepteurs de journal requis, utilisez le programme d'exit qui supprime le récepteur de journal qui a été automatiquement enregistré lorsque vous avez installé DB2 DataPropagator pour System i. Ce programme d'exit est appelé à chaque fois que vous ou l'un de vos programmes d'application tente de supprimer un récepteur de journal. Ce programme d'exit détermine ensuite si un récepteur de journal peut ou non être supprimé.

**Recommandation :** Indiquez DLTRCV(\*YES) et MNGRCV(\*SYSTEM) sur la commande **CHGJRN** ou **CRTJRN** pour utiliser le programme d'exit qui supprime un récepteur de journal et laisser la gestion du journal sur le système.

Si le récepteur de journal est utilisé par une ou plusieurs tables source, le programme d'exit qui supprime un récepteur de journal vérifie que le récepteur supprimé ne contient pas d'entrées n'ayant pas été traitées par le programme Capture. Le programme d'exit *désapprouve* la suppression du récepteur si le programme Capture doit encore traiter des entrées sur ce récepteur.

## Éléments à prendre en compte pour la gestion des dictionnaires de compression (z/OS)

Si vous utilisez des utilitaires de dictionnaires de compression DB2, vous devez coordonner l'utilisation de ces utilitaires avec vos programmes Capture.

### Mise à jour des dictionnaires de compression DB2 (z/OS)

Lorsque le programme Capture demande des enregistrements de journal, DB2 doit décompresser les enregistrements de journal de toute table qui est stockée dans un espace table compressé. Pour la décompression, DB2 utilise le dictionnaire de compression en cours. Dans certains cas, le dictionnaire de compression peut être indisponible. Le programme Capture prend différentes mesures dans chaque cas :

#### Si le dictionnaire de compression est provisoirement indisponible

DB2 renvoie une erreur au programme Capture. Le programme Capture fait plusieurs tentatives pour continuer le traitement. Si le dictionnaire demeure indisponible, le programme Capture émet un message ASN0011E et se termine.

#### Si le dictionnaire de décompression est indisponible de manière définitive

Un dictionnaire de compression peut être perdu si vous utilisez REORG sans indiquer KEEPDICTIONARY=YES. Dans ce cas, le programme Capture suit l'action de l'erreur spécifié par l'option STOP\_ON\_ERROR pour l'enregistrement. Si STOP\_ON\_ERROR=N (no), Capture désactive l'enregistrement. Si STOP\_ON\_ERROR=Y (yes), le programme Capture émet un message ASN0011E et se termine.

Avec APAR PK19539 (DB2 for 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 utilisez deux fois l'utilitaire REORG pour le même espace table avant que le programme Capture ne lise tous les anciens enregistrements de journal pour cette table.

Pour éviter ces situations dans DB2 for z/OS version 7, laissez le programme Capture traiter tous les enregistrements de journal pour une table avant d'effectuer toute activité qui affecte 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 (z/OS)

Vous devez également envisager la disponibilité de votre dictionnaire de compression. Lorsque le programme Capture lit les enregistrements de journal compressés, DB2 prend un verrou sur l'espace table compressé de la source pour accéder au dictionnaire. Le programme Capture s'arrête si l'espace table compressé sur le système source est dans l'état STOPPED

lorsque l'interface de lecture de journal de DB2 a besoin de ce verrou. A 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 Capture pendant qu'il lit le dictionnaire.


Pour éviter tout verrouillage temporaire dû à un verrou indisponible, suspendez le programme Capture lorsqu'un espace table compressé de la source doit être utilisé exclusivement par un utilitaire DB2 (ou fournisseur).

---

## Maintenance des tables de contrôle

La réplication SQL utilise des tables de contrôle pour conserver les définitions source, les définitions d'ensembles d'abonnements et d'autres informations de contrôle propres à la réplication. Bien que la taille de certaines tables de contrôle soit statique, d'autres tables de contrôle peuvent s'agrandir puis se réduire de façon dynamique, en fonction de la taille de votre base de données et de vos besoins en matière de réplication.

La taille des tables suivantes change fréquemment au cours d'un traitement normal :

-  IBMSNAP\_APPLY\_JOB
- IBMSNAP\_APPLYTRACE
- IBMSNAP\_APPLYTRAIL
- IBMSNAP\_CAPMON
- IBMSNAP\_CAPTRACE
- tables de modification des données
- tables CCD
- IBMSNAP\_ALERTS
- IBMSNAP\_MONTRACE
- IBMSNAP\_MONTRAIL
- IBMSNAP\_SIGNAL
- BMSNAP\_SUBS\_EVENT
- IBMSNAP\_UOW

La taille et la croissance de ces tables de contrôle dynamiques sont susceptibles d'affecter les performances de votre système.

## Utilitaire RUNSTATS pour la réplication SQL (Linux, UNIX, Windows, z/OS)

L'utilitaire RUNSTATS met à jour les statistiques concernant les caractéristiques physiques de vos tables et index associés.

Vous devez continuer d'exécuter l'utilitaire RUNSTATS sur vos tables existantes avec la même fréquence que lorsque vous utilisiez la réplication SQL. Toutefois, n'exécutez qu'une fois l'utilitaire RUNSTATS sur les tables CD, sur les tables IBMSNAP\_UOW et sur les autres tables de contrôle dynamiques 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.

## Lier à nouveau les modules et plans (z/OS, Linux, UNIX, Windows)

La redéfinition de modules et de plans avec des lectures non validées d'isolement maintient les performances du système à un niveau optimal.

De nombreux modules et plans de réplication SQL sont redéfinis avec des lectures non validées d'isolement. Si vous devez redéfinir vos modules et vos plans, notez que vos programmes de maintenance interne utilisés pour la redéfinition automatique risquent d'entraîner des conflits entre le programme Capture et le programme Apply si ces programmes redéfinissent les modules de réplication à l'aide des options standard telles que la lecture non reproductible. Les modules de réplication SQL doivent conserver les lectures non validées d'isolement pour que les performances du système soient optimales.

## Réorganisation des tables de contrôle

Vous devez réorganiser régulièrement les tables de contrôle dynamique qui font l'objet de fréquentes mises à jour.

### A propos de cette tâche

Vos tables CD et IBMSNAP\_UOW font l'objet de nombreuses opérations INSERT au cours de la capture des modifications et de nombreuses opérations DELETE au cours de l'élagage. La taille des tables IBMSNAP\_CAPMON, IBMSNAP\_CAPTRACE et IBMSNAP\_APPLYTRAIL peut varier considérablement selon l'importance des mises à jour effectuées dans les tables sources de réplication.

**Recommandation :** Réorganisez les tables de contrôle dynamiques suivantes une fois par semaine :

- tables de modification des données
- IBMSNAP\_ALERTS
- IBMSNAP\_APPLYTRACE
- IBMSNAP\_APPLYTRAIL
- IBMSNAP\_CAPMON
- IBMSNAP\_CAPTRACE
- IBMSNAP\_MONTRAIL
- IBMSNAP\_MONTRACE
- IBMSNAP\_UOW

Pour les autres tables de contrôle, il est inutile d'exécuter des utilitaires qui récupèrent l'espace inutilisé ou de générer des statistiques de l'optimiseur fréquemment mises à jour.

### Procédure

Pour réorganiser les tables de contrôle, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>z/OS</b> Utilitaire REORG avec l'option PREFORMAT	L'option PREFORMAT de cet utilitaire accélère le traitement d'insertion du programme Capture.

Méthode	Description
<p><b>System i</b></p> <p>Commande <b>RGZPFM</b> (réorganisation de membre de fichier physique)</p>	Vous pouvez réorganiser la table UOW et les tables CD actives lorsque le programme Capture se termine en spécifiant le paramètre <b>RGZCTLTBL(*YES)</b> pour la commande <b>ENDDPRCAP</b> .
<p><b>Linux UNIX Windows</b></p> <p>Commande <b>REORG</b></p>	Utilisez cette commande pour éliminer les données fragmentées et récupérer de l'espace.

## Elagage des tables de contrôle dynamiques gérées par les programmes Capture (Linux, UNIX, Windows, z/OS)

Vous pouvez élaguer manuellement ou automatiquement les tables dont la taille fluctue.

### A propos de cette tâche

Vous devez surveiller la croissance des tables de contrôle dynamiques suivantes et évaluer les différentes méthodes d'élagage proposées :

- tables de modification des données
- IBMSNAP\_UOW
- IBMSNAP\_CAPMON
- IBMSNAP\_CAPTRACE
- IBMSNAP\_SIGNAL
- **System i** IBMSNAP\_AUTHTKN

Vous pouvez configurer les programmes Capture en vue de l'élagage automatique de ces tables à intervalles réguliers. Vous pouvez également effectuer un élagage sur demande en lançant une première fois le processus d'élagage : le programme Capture n'effectue pas de nouvel élagage tant que vous n'avez pas entré la commande d'élagage suivante.

### Procédure

Pour élaguer des tables de contrôle dynamiques gérées par le programme Capture, procédez comme suit :

1. Si vous souhaitez élaguer automatiquement les tables de contrôle dynamiques, affectez au paramètre **autoprune** la valeur **yes** en utilisant l'une des méthodes suivantes :

Méthode	Description
Lancez un programme Capture avec élagage automatique.	Emettez la commande système <b>asncap</b> en spécifiant <b>autoprune=y</b> . Définissez le paramètre <b>prune_interval</b> afin d'indiquer la fréquence d'élagage automatique.
Activez l'élagage automatique pour le programme Capture en cours de fonctionnement.	Emettez la commande système <b>asnccmd chgparms</b> en spécifiant <b>autoprune=y</b> . Définissez le paramètre <b>prune_interval</b> afin d'indiquer la fréquence d'élagage automatique.

2. Si vous souhaitez élaguer une seule fois les tables de contrôle dynamiques, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Centre de réplication</b>	Utilisez la fenêtre Elaguer les tables de contrôle Capture pour effectuer un seul élagage des tables de contrôle. Pour ouvrir cette fenêtre, affichez la branche <b>Opérations</b> de l'arborescence des objets et cliquez sur le dossier <b>Serveurs de contrôle de Capture</b> ; ensuite, cliquez avec le bouton droit de la souris sur un serveur dans le panneau de contenu et sur <b>Elaguer Capture</b> .
<b>Effectuez un élagage dans un programme Capture en cours de fonctionnement.</b>	Emettez la commande système <b>asncmd</b> en spécifiant le paramètre d'élagage.

## Elagage des tables de modification des données (CD) et des tables d'unités de travail (UOW)

Au cours de chaque cycle d'élagage (appelé automatiquement ou sur demande), le programme Capture effectue l'élagage des tables CD et UOW sur la base de la progression indiquée par les programmes Apply.

La progression de l'élagage est signalée par la valeur de colonne SYNCHPOINT de la table IBMSNAP\_PRUNE\_SET. Cet élagage normal s'appuie sur la valeur minimale de point de synchronisation de tous les programmes Apply abonnés à chaque table CD et sur la valeur minimale de point de synchronisation globale de la table UOW.

Cependant, l'élagage normal ne permet pas d'élaguer efficacement les tables CD et UOW si les abonnements associés ne sont exécutés qu'occasionnellement. Lorsque vous déterminez la fréquence d'exécution des programmes Apply associés, lorsque vous arrêtez ces programmes Apply et lorsque vous désactivez les ensembles d'abonnements pendant une période relativement longue, tenez compte de l'efficacité de l'élagage.

Si vous exécutez vos ensembles d'abonnements très rarement ou que vous arrêtez les programmes Apply, la taille de vos tables CD et UOW peut croître énormément ; ces tables deviennent alors candidates pour l'élagage de durée de conservation. La durée de conservation est un paramètre opérationnel du programme Capture, qui porte une valeur par défaut d'une semaine. Elle détermine la durée de conservation des anciennes données dans les tables avant d'être candidates pour l'élagage de limite de conservation.

Si le processus d'élagage normal ne peut pas se faire pour cause d'ensembles d'abonnements désactivés ou exécutés peu souvent, les données sont conservées dans les tables pendant de longues périodes. Lorsque ces données sont plus anciennes que l'horodatage DB2 actuel moins la valeur de durée de conservation, le processus d'élagage de durée de conservation supprime ces données des tables.

Essayez d'éviter les élagages de durée de conservation, car l'accumulation de données anciennes peut entraîner un dépassement de capacité de stockage et une diminution des performances.

**Recommandation :** Exécutez vos programmes Apply une fois par jour minimum, pour tous vos ensembles d'abonnements.

Si le serveur source fournit des données modifiées à un grand nombre de systèmes cible, avec chaque fois des configurations différentes et parfois une exécution peu fréquente des programmes Apply pour les sources enregistrées, l'utilisation de plusieurs programmes Capture est à envisager. Vous pouvez utiliser plusieurs programmes Capture et gérer les différentes conditions de traitement requises à l'aide de schémas Capture différents : un schéma Capture utilisé pour isoler les tables élaguées peu souvent en raison d'exigences spécifiques au niveau des ensembles d'abonnements, et un autre schéma Capture pour les autres tables source.

## **Recommandations relatives à l'élagage d'autres tables de contrôle dynamiques**

Vous devez procéder régulièrement à l'élagage de vos tables de contrôle de réplication, afin de supprimer les données obsolètes et d'accroître les performances système.

Le programme Capture effectue des opérations d'élagage pour les tables qu'il gère uniquement. Le programme Apply gère des tables de modification cohérente des données (CCD) ; par conséquent, le programme Capture n'effectue pas d'élagage automatique de ces tables. Certains types de tables CCD n'ont pas besoin d'élagage. Les tables CCD condensées font l'objet de mises à jour.

Les seuls enregistrements à supprimer de ces tables sont ceux dont la valeur D (Delete) de la colonne IBMSNAP\_OPERATION a déjà été répliquée dans les tables cible associées. Les tables CCD non condensées contiennent des données d'historique et peuvent devenir très volumineuses. Vous devez conserver ces données à des fins d'audit ; par conséquent, vous ne devez pas effectuer d'élagage sur les tables CCD non condensées.

Toutefois, vous devez envisager l'élagage de vos tables CCD internes. En effet, ces tables connaissent une croissance rapide si de fréquentes mises à jour ont lieu sur votre système. Seules les modifications les plus récentes sont extraites des tables CCD internes ; par conséquent, il est inutile de conserver les lignes plus anciennes.

Pour activer l'élagage des tables CCD internes, vous devez ajouter des instructions SQL aux ensembles d'abonnements associés, afin d'élaguer les données déjà appliquées à toutes les cibles concernées. Vous avez également la possibilité d'ajouter les instructions SQL DELETE requises à vos fonctions de planification automatique, afin de supprimer les lignes souhaitées de ces tables.

Vous devez également élaguer manuellement les tables IBMSNAP\_APPLYTRAIL et IBMSNAP\_APPLYTRACE. Si vous définissez et utilisez plusieurs ensembles d'abonnements avec des programmes Apply fréquemment exécutés, la taille de la table IBMSNAP\_APPLYTRAIL augmente rapidement et nécessite des élagages fréquents. Pour gérer la croissance de ces tables, la meilleure méthode consiste à ajouter une instruction SQL ou un appel de procédure à l'un des ensembles d'abonnements utilisés. Vous pouvez également ajouter une instruction SQL DELETE à vos fonctions de planification automatique.

## **Prévention des échecs de réplication et reprise après erreur**

Ces rubriques décrivent les méthodes de prévention des échecs de réplication affectant vos tables de contrôle et vos données de réplication, ainsi que les méthodes de reprise après la survenue d'un échec.

## Préventions des démarrages à froid du programme Capture

Vous ne devez effectuer un démarrage à froid du programme Capture que si vous démarrez le programme pour la première fois ou que vous avez besoin d'effectuer une régénération intégrale de vos tables de contrôle et de vos tables cible. Si vous démarrez le programme Capture à froid, toutes les tables cible de votre environnement de réplication sont régénérées.

**z/OS** **Linux UNIX Windows** Lorsqu'un programme Capture démarre à l'aide de l'option warmns ou warmsi, il tente d'extraire les enregistrements de journal sur la base du point de reprise indiqué dans la table IBMSNAP\_RESTART. Si le programme Capture ne parvient pas à trouver le journal, son démarrage à chaud échoue.

Pour prévenir un démarrage à froid du programme Capture, tenez compte des considérations suivantes :

- **System i** Démarrez le programme Capture à l'aide du paramètre RESTART(\*YES). Le programme Capture poursuit le traitement à partir du point où il se trouvait lorsqu'il s'est arrêté précédemment. Conservez une quantité suffisante de données de journal ou de récepteurs de journaux DB2 sur votre système et assurez-vous que ces données sont disponibles pour la réplication SQL.
- Utilisez le moniteur d'alertes de réplication ou un autre système pour vérifier le statut des données d'historique de vos programmes Capture. Vous pouvez ensuite utiliser ces informations pour vérifier que les programmes Capture sont toujours en fonctionnement lorsque DB2 est actif.
- Assurez-vous que vous conservez une quantité suffisante de données de journal ou de récepteurs de journaux DB2 sur votre système, et que ces données sont disponibles pour la réplication SQL.

## Reprise après une erreur E-S et un échec de connectivité dans les tables de contrôle

Si la réplication perd la connectivité à une table de contrôle, vous pouvez effectuer une reprise ; pour les autres erreurs, le programme de réplication s'arrête.

### A propos de cette tâche

Si le programme Capture détecte une erreur E-S ou un échec de connectivité, il génère le message d'erreur correspondant et s'arrête.

Le programme Apply s'arrête s'il détecte des erreurs catastrophiques au niveau des tables de contrôle. S'il détecte des erreurs dans les tables cible ou des erreurs de connectivité réseau, il consigne l'erreur dans la table IBMSNAP\_APPLYTRAIL, puis continue de fonctionner.

### Procédure

Pour obtenir une reprise après la survenue d'erreurs et d'échecs de connectivité dans les tables de contrôle, procédez comme suit :

1. En cas d'erreur E-S ou d'échec de connectivité dans une table de contrôle, utilisez une procédure de reprise DB2 standard pour récupérer la table. Dans ce cas, aucune donnée de la table n'est perdue.
2. Si le programme s'arrête, redémarrez le programme Capture à partir du point de défaillance, puis relancez le programme Apply.



## Récupération des données source perdues

Si vous perdez des données sources, vous pouvez les récupérer via une méthode de point de récupération ou via l'exécution d'une régénération intégrale.

### A propos de cette tâche

Si une table source est récupérée au point de défaillance, la réplication SQL se poursuit normalement. Une fois la table récupérée, le programme Capture poursuit la collecte des modifications de données pour cette table.

Toutefois, les programmes Capture et Apply ne détectent pas une récupération de point de cohérence au sein d'une table cible en lecture seulement. Si vous récupérez une table source, le programme Apply peut avoir répliqué les modifications dans les tables cible qui n'existent plus au niveau de la source, ce qui provoque des incohérences entre vos tables source et vos tables cible, si vous ne parvenez pas à faire revenir les tables cible au même point de cohérence logique.

Ce scénario devient encore plus complexe en présence de plusieurs niveaux de réplication. Vous devez soit développer un système qui fournisse des points de récupération correspondants parmi les différents niveaux, soit utiliser une régénération intégrale, selon vos préférences.

### Procédure

Pour récupérer vos données source, utilisez l'une des méthodes suivantes :

Méthode	Description
<b>Système de point de récupération</b>	Vous devez développer un système qui fournisse des points de récupération correspondants parmi les différents niveaux de réplication.
<b>Régénération intégrale</b>	Vous devez utiliser une régénération intégrale comme méthode de récupération choisie

## Elagage des tables IBMSNAP\_CAPMON et IBMSNAP\_CAPTRACE

Les valeurs affectées aux paramètres d'exploitation déterminent l'élagage des tables IBMSNAP\_CAPMON et IBMSNAP\_CAPTRACE.

Au cours de chaque cycle d'élagage, le programme Capture effectue l'élagage des tables IBMSNAP\_CAPMON et IBMSNAP\_CAPTRACE sur la base des valeurs affectées aux paramètres d'exploitation suivants du programme Capture :

- Le paramètre **monitor\_limit**(Linux, UNIX, Windows, z/OS) et le paramètre **MONLMT**(System i) déterminent la durée de conservation des lignes dans la table IBMSNAP\_CAPMON
- Le paramètre **trace\_limit**(Linux, UNIX, Windows, z/OS) et le paramètre **TRCLMT**(System i) déterminent la durée de conservation des lignes dans la table IBMSNAP\_CAPTRACE

Les paramètres de limite de moniteur et de limite de trace portent tous une valeur par défaut égale à une semaine. Vous pouvez modifier ces valeurs en fonction de la durée nécessaire de conservation des informations de rendement et de latence Capture dans la table IBMSNAP\_CAPMON et de conservation des données d'audit et d'identification des incidents dans la table IBMSNAP\_CAPTRACE.

## Élagage de la table IBMSNAP\_SIGNAL

La table IBMSNAP\_SIGNAL est élaguée automatiquement, car des lignes sont constamment ajoutées au cours de la réplication.

La table IBMSNAP\_SIGNAL est également élaguée au cours de chaque cycle d'élagage. Une ligne de signal est admissible pour l'élagage lorsque la valeur de la colonne SIGNAL\_STATE est égale à C. La valeur C indique que les informations sur le signal sont complètes et qu'elles ne sont plus requises au niveau du programme Capture ou au niveau de tout autre traitement utilisateur, et qu'elles sont admissibles pour l'élagage. Une ligne de signal portant dans la colonne SIGNAL\_TIME une valeur antérieure à celle de l'horodatage DB2 en cours moins la valeur du paramètre de limite de conservation est admissible pour l'élagage de limite de conservation.

---

## Gestion 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, qu'il s'agisse de tables de base de données existantes ou de tables que vous avez spécifiées pour être générées automatiquement par la réplication SQL.

**Remarque :** Désactivez vos programmes Apply avant de passer une table cible hors ligne afin d'exécuter un utilitaire.

---

## Chapitre 14. 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 224
- «Quand utiliser l'utilitaire de comparaison», à la page 224
- «Stockage des différences», à la page 225
- «Autorisations obligatoires», à la page 226
- «Restrictions concernant les colonnes clés dans la source et la cible», à la page 226
- «Considérations relatives au type de données», à la page 226
- «Effets du filtrage», à la page 227
- «Comparaisons basées sur les requêtes au lieu des abonnements», à la page 228
- «Comparaison d'un sous-ensemble de lignes de table», à la page 229

## 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 301.

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 230
- «Autorisations obligatoires», à la page 230
- «Restrictions», à la page 231
- «Conseils d'utilisation», à la page 231

## 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 301.

## 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 233
- «Comparaison de colonnes clé TIMESTAMP dotées de précisions différentes», à la page 233
- «Éléments à prendre en compte lors de l'utilisation des types de données DATE et TIMESTAMP(0)», à la page 235

- «Comportement lors de l'utilisation du paramètre rangecol », à la page 234
- «Option de compatibilité pour les chaînes de texte », à la page 235
- «Option de fichier asntdiff (asntdiff -f) », à la page 236

### 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 - <b>TIMESTAMP</b> (6) 2009-02-05-12.46.01.126412	Co11 - <b>TIMESTAMP</b> (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 - <b>TIMESTAMP</b> (12) 2009-02-05-12.46.01.126412123456	Co11 - <b>TIMESTAMP</b> (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 - <b>TIMESTAMP</b> (6) 2009-02-05-12.46.01.126412	KEYCo11 - <b>TIMESTAMP</b> (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 15. Planification de programmes de réplication SQL sur différents systèmes d'exploitation

Vous pouvez planifier le démarrage du programme Capture, du programme Apply ou du moniteur d'alertes de réplication à une heure précise, à l'aide des commandes disponibles sur votre système d'exploitation.

---

### Planification de programmes sous Linux et UNIX

Vous pouvez planifier le démarrage des programmes de réplication sous Linux et UNIX.

#### Procédure

Pour planifier les programmes de réplication sous Linux et UNIX, procédez comme suit :

Utilisez la commande **at** pour démarrer un programme de réplication à un moment spécifique. Le tableau 17 affiche les commandes utilisées pour démarrer les programmes de réplication à 15:00 le vendredi :

Tableau 17. Planification des commandes pour les programmes de réplication (Linux, UNIX)

Programme de réplication	Commande Linux ou UNIX
Capture	at 3pm Friday asncap autoprune=n
Apply	at 3pm Friday asnapply applyqual=myqual
moniteur d'alertes de réplication	at 3pm Friday asnmon monitor_server=db2srv1 monitor_qualifier=mymon

---

### Planification de programmes sous Windows

Vous pouvez planifier le démarrage des programmes de réplication sur les systèmes d'exploitation Windows.

#### Procédure

Si vous n'utilisez pas le gestionnaire de contrôle de service Windows, utilisez la commande **AT** pour démarrer les programmes à un moment précis. Avant d'entrer la commande **AT**, démarrez le service de planification Windows. Le tableau 18 affiche les commandes utilisées pour démarrer les programmes de réplication à 15 h 00 le vendredi :

Tableau 18. Planification des commandes pour les programmes de réplication (Windows)

Programme de réplication	Commande Windows
Capture	c:\>at 15:00/interactive"c:\SQLLIB\BIN\db2cmd.exe c:\CAPTURE\asncap.exe"
Apply	c:\>AT 15:00 /interactive "c:\SQLLIB\BIN\db2cmd.exe c:\SQLLIB\BIN\asnapply.exe control_server=cntldb apply_qual=qualid1"

Tableau 18. Planification des commandes pour les programmes de réplication (Windows) (suite)

Programme de réplication	Commande Windows
moniteur d'alertes de réplication	c:\>AT 15:00 /interactive "c:\SQLLIB\BIN\db2cmd.exe c:\CAPTURE\asnmon.exe monitor_server=db2srv1 monitor_qualifier=mymon"

---

## Planification des programmes sur les systèmes d'exploitation z/OS

Vous pouvez planifier le lancement des programmes de réplication sur le systèmes d'exploitation z/OS à l'aide de deux commandes différentes.

### Procédure

Pour planifier les programmes sur les systèmes d'exploitation z/OS, utilisez les méthodes suivantes :

1. Créez une procédure qui appelle le programme pour z/OS dans la PROCLIB.
2. Modifiez le module ICHRIN03 RACF (ou les définitions appropriées pour votre module 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 Capture ou le programme Apply à un moment spécifique. Voir les *commandes MVS/ESA JES2* pour plus d'informations sur l'utilisation de la commande **\$TA JES2**. Voir *NetView for MVS Command Reference* pour obtenir plus d'informations sur l'utilisation de la commande **AT NetView**.

---

## Planification des programmes sur le système d'exploitation System i

Vous pouvez planifier le moment du démarrage des programmes de réplication sur le système d'exploitation System i.

### Procédure

1. Si vous souhaitez démarrer le programme Apply, émettez la commande **ADDJOBSCDE**.
2. Si vous souhaitez démarrer le programme Capture, émettez la commande **SBMJOB**. Par exemple :  
SBMJOB CMD('STRDPRCAP...')SCDDATE(...)SCDTIME(...)

---

## Chapitre 16. 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 19 décrit les entrées pour le nom du service de réplication.

Tableau 19. 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 :

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 **ansdrop**.

---

## Chapitre 17. Communication entre les composants de la réplication SQL

Les différents composants de réplication sont exécutés de manière indépendante, mais sont interdépendants au niveau des informations que chacun d'entre eux enregistre dans les tables de contrôle de réplication en vue d'établir la communication.

### Outils d'administration

Le Centre de réplication ou le programme de ligne de commandes ASNCLP crée des scripts SQL qui insèrent les informations initiales sur les sources enregistrées, sur les ensembles d'abonnements et sur les critères d'alerte dans les tables de contrôle.

### Programme ou déclencheurs Capture

Le programme et les déclencheurs Capture mettent à jour les tables de contrôle afin de refléter l'avancement de la réplication et de coordonner le traitement des modifications.

### Programme Apply

Le programme Apply met à jour les tables de contrôle afin de refléter l'avancement de la réplication et de coordonner le traitement des modifications.

### Moniteur d'alertes de réplication

Le moniteur d'alertes de réplication lit les tables de contrôle mises à jour par le programme Capture, par le programme Apply et par les déclencheurs Capture, afin de comprendre les incidents survenus et d'identifier la progression sur un serveur.

---

## Centre de réplication, ASNCLP, programme ou déclencheurs Capture et programme Apply

Lorsque vous enregistrez une table, une vue ou un pseudonyme en tant que source de réplication, le Centre de réplication ou le programme de ligne de commandes ASNCLP crée un script SQL qui stocke les informations de cette source dans la table de contrôle de réplication contenant toutes les informations d'enregistrement : la table IBMSNAP\_REGISTER. Le script SQL généré par les outils d'administration crée également les tables de modification des données (CD) pour les sources enregistrées.

La table IBMSNAP\_REGISTER contient une ligne par table de source enregistrée et une ligne par table sous-jacente dans une vue enregistrée. Cette table contient les types suivants d'informations sur chaque source enregistrée :

- Nom du schéma et nom de la table source
- Type de structure de chaque table de source enregistrée
- Nom du schéma et nom de la table de modification des données (CD)
- Noms des tables de modification des données (CD) des tables sous-jacentes de cette vue (uniquement pour les vues enregistrées, et uniquement si les tables sous-jacentes sont enregistrées)
- Nom du schéma et nom de la table CCD interne (le cas échéant)
- Niveau de détection de conflit pour les sources de réplication bidirectionnelle

Les programmes Capture et Apply utilisent les informations contenues dans la table IBMSNAP\_REGISTER pour se communiquer leur statut respectif. Cette table possède plusieurs autres colonnes destinées aux informations associées.

Pour les sources System i, y compris les tables journalisées à distance, il existe également une extension de la table IBMSNAP\_REGISTER (la table IBMSNAP\_REG\_EXT), qui contient des informations supplémentaires propres aux systèmes System (nom de la bibliothèque du journal et nom du journal, par exemple).

Lorsque vous créez un ensemble d'abonnements et que vous lui ajoutez des membres, le Centre de réplication crée un script SQL qui stocke les informations de cet ensemble d'abonnements dans les tables de contrôle de réplication contenant toutes les informations relatives à cet ensemble, comme indiqué ci-après :

- Table IBMSNAP\_SUBS\_SET
- Table IBMSNAP\_SUBS\_MEMBR
- Table IBMSNAP\_SUBS\_COLS
- Table IBMSNAP\_SUBS\_STMTS

Si les tables cible n'existent pas encore, le script SQL généré par le Centre de réplication les crée.

La table principale d'ensembles d'abonnements (IBMSNAP\_SUBS\_SET) contient une ligne par ensemble d'abonnements. Cette table contient les types d'informations suivants sur chaque ensemble d'abonnements :

- Qualificatif Apply
- Nom de l'ensemble d'abonnements
- Type d'ensemble d'abonnements : en lecture seule ou en lecture/écriture (réplication bidirectionnelle)
- Noms et alias des bases de données source et cible
- Heure de traitement de l'ensemble d'abonnements
- Etat actuel de l'ensemble d'abonnements

Cette table possède également plusieurs autres colonnes consacrées aux informations associées.

Les autres tables d'ensembles d'abonnements contiennent des informations sur les membres, colonnes et instructions SQL (ou procédures stockées) d'ensembles d'abonnements traités avec ces derniers.

---

## Programme Capture et programme Apply

Le programme Capture utilise certaines tables de contrôle de réplication pour indiquer que des modifications ont été apportées à la base de données source ; le programme Apply utilise ces valeurs de table de contrôle pour détecter les éléments à copier dans la base de données cible.

Le programme Capture ne capture pas les informations tant que les signaux du programme Apply ne lui ont pas indiqué de le faire, et le programme Apply n'envoie pas de signaux au programme Capture pour le démarrage de la capture des modifications tant que vous n'avez pas défini de source de réplication et d'ensembles d'abonnements associés.



La liste ci-après décrit la façon dont le programme Capture et le programme Apply communiquent dans le cadre d'un scénario de réplication classique, pour garantir l'intégrité des données :

#### **Capture des données à partir d'une base de données source**

1. Le programme Capture lit la table IBMSNAP\_REGISTER pendant le démarrage, pour identifier les sources de réplication enregistrées pour lesquelles il doit capturer les modifications. Ensuite, il conserve les informations d'enregistrement en mémoire.
2. Le programme Capture lit le journal DB2 en continu, afin de détecter des enregistrements de modification (opérations INSERT, UPDATE et DELETE) pour les tables (ou vues) source enregistrées. Il détecte également les insertions effectuées dans la table IBMSNAP\_SIGNAL, afin d'obtenir les actions initialisées par le programme Apply ou par un utilisateur. Lorsque le programme Apply insère un signal CAPSTART dans la table IBMSNAP\_SIGNAL et que le programme Capture détecte ce signal, il initialise l'enregistrement et commence la capture des modifications pour la source associée.
3. Une fois que le programme Capture a commencé la capture des modifications pour une source enregistrée, il insère une ligne (ou deux lignes si vous avez spécifié que les mises à jour devaient être enregistrées sous forme d'instructions DELETE et INSERT) dans la table CD de chaque modification validée trouvée dans le journal DB2. Le programme Capture conserve les modifications non validées en mémoire, jusqu'à ce qu'elles soient validées ou abandonnées. Chaque source de réplication enregistrée ne constituant pas une table CCD externe possède une table CD associée.
4. A chaque intervalle de validation, le programme Capture valide les données enregistrées dans les tables CD et CCD, et met à jour la table IBMSNAP\_REGISTER pour signaler les tables CD dont les modifications ont été validées.

#### **Application des données à une base de données cible**

1. Pour tous les ensembles d'abonnements définis récemment, le programme Apply demande tout d'abord aux déclencheurs Capture de commencer la capture des données modifiées. Ensuite, une régénération intégrale est effectuée pour chaque membre de l'ensemble (sauf en cas de table cible incomplète).
2. Lorsqu'un ensemble d'abonnements est admissible pour la réplication, le programme Apply examine la table IBMSNAP\_REGISTER pour déterminer si des modifications doivent être répliquées.
3. Le programme Apply copie les modifications de la table CD dans la table cible.
4. Le programme Apply met à jour la table IBMSNAP\_SUBS\_SET pour enregistrer les données qu'il a copiées pour chaque ensemble d'abonnements.
5. Le programme Apply met à jour la table IBMSNAP\_PRUNE\_SET à l'aide d'une valeur indiquant le point jusqu'auquel il a lu les modifications apportées à la table CD.

#### **Elagage des tables CD**

Lorsque le programme Capture effectue l'élagage des tables CD, il utilise les informations situées dans la table IBMSNAP\_PRUNE\_SET pour

déterminer les modifications qui ont été appliquées et supprime les modifications qui ont déjà été répliquées à partir de la table de modification des données (CD).

---

## Déclencheurs Capture et programme Apply

Les déclencheurs Capture utilisent certaines tables de contrôle de réplication pour indiquer que des modifications ont été apportées à la base de données source ; le programme Apply utilise ces valeurs de table de contrôle pour détecter les éléments à copier dans la base de données cible.

Les déclencheurs Capture commencent immédiatement la capture des informations. Contrairement au programme Capture, ils n'attendent pas l'émission d'un signal en provenance du programme Apply.

La liste ci-après décrit la façon dont le programme Capture et le programme Apply communiquent dans le cadre d'un scénario de réplication classique, pour garantir l'intégrité des données :

### Capture des données à partir d'une source

1.

Chaque fois qu'une opération DELETE, UPDATE ou INSERT a lieu dans la table source de réplication, un déclencheur Capture enregistre la modification dans la table CCD pour cette table source.

### Application des données à une cible

1. Pour tous les ensembles d'abonnements définis récemment, le programme Apply demande tout d'abord aux déclencheurs Capture d'insérer un point de départ valide dans la table CCD marquant le début de la recherche des données modifiées. Ensuite, une actualisation complète est effectuée pour chaque membre de l'ensemble (sauf en cas de table cible incomplète).
2. Lorsque le programme Apply traite un ensemble d'abonnements pour une source relationnelle non DB2, il met à jour la table IBMSNAP\_REG\_SYNCH, ce qui active un déclencheur UPDATE pour cette table. Le déclencheur met à jour la valeur SYNCHPOINT de la table IBMSNAP\_REGISTER, pour représenter la valeur SYNCHPOINT la plus élevée des tables CCD copiées dans les cibles. Au cours du cycle suivant, le programme Apply traite les nouvelles données figurant dans la table CCD possédant une valeur SYNCHPOINT inférieure ou égale à cette valeur SYNCHPOINT. La table IBMSNAP\_REG\_SYNCH ne se trouve pas dans la base de données non DB2 ; par conséquent, le programme Apply utilise un pseudonyme créé par le Centre de réplication pour effectuer les enregistrements dans la table.
3. Le programme Apply examine la table IBMSNAP\_REGISTER pour déterminer si des modifications doivent être répliquées.
4. Le programme Apply copie les modifications de la table CCD dans la table cible.
5. Le programme Apply met à jour la table IBMSNAP\_SUBS\_SET pour enregistrer les données qu'il a copiées pour chaque ensemble d'abonnements.
6. Le programme Apply met à jour la table IBMSNAP\_PRUNCNTL pour chaque source enregistrée à l'aide d'une valeur indiquant le point jusqu'auquel il a lu les modifications de la table CCD.

### Elagage des tables CCD

Le déclencheur UPDATE de la table IBMSNAP\_PRUNCNTL vérifie toutes les tables CCD présentes dans la base de données source et supprime les modifications déjà répliquées des tables CCD correspondantes.

---

## Outils d'administration et moniteur d'alertes de réplication

Lorsque vous définissez une alerte avec les contacts qui seront notifiés en cas de survenue de cette situation, le Centre de réplication ou le programme ASNCLP crée un script SQL qui stocke les informations de cette alerte dans les tables de contrôle de réplication contenant toutes les informations sur l'alerte et sur la notification.

Les tables de contrôle suivantes sont mises à jour :

- Table IBMSNAP\_CONDITIONS
- Table IBMSNAP\_CONTACTS
- Table IBMSNAP\_GROUPS
- Table IBMSNAP\_CONTACTGRP

Les tables IBMSNAP\_CONDITIONS contiennent une ligne par situation d'alerte à surveiller. Cette table contient les types suivants d'informations sur chaque condition d'alerte enregistrée :

- Qualificatif du moniteur
- Nom et alias du serveur Capture ou du serveur Apply à surveiller
- Composant à surveiller (programme Capture ou programme Apply)
- Schéma Capture ou qualificatif Apply
- Nom de l'ensemble d'abonnements défini (si vous souhaitez surveiller un ensemble)
- Condition d'alerte à surveiller
- Contact à notifier si cette condition d'alerte se déclenche

Cette table possède plusieurs autres colonnes destinées aux informations associées.

Les autres tables du moniteur d'alertes de réplication contiennent des informations sur la personne à notifier si la condition d'alerte se produit (contact individuel ou groupe de contacts, ou encore la console z/OS), mode de notification (courrier électronique ou récepteur d'appel), ainsi que la fréquence de notification du contact en cas de prolongement de la situation surveillée.

---

## Moniteur d'alertes de réplication, programme Capture et programme Apply

Le moniteur d'alertes de réplication utilise certaines tables de contrôle Capture pour surveiller le programme Capture, ainsi que certaines tables de contrôle Apply pour surveiller le programme Apply. Il effectue une lecture à partir de différentes tables de contrôle de réplication sur chaque serveur de contrôle Capture ou Apply, en fonction de l'élément surveillé.

Le moniteur d'alertes de réplication ne communique pas avec le programme Capture ou Apply.

La procédure suivante décrit la méthode de surveillance, par le moniteur d'alertes de réplication, des critères applicables aux programmes Capture et Apply, et notifie les contacts lorsqu'un critère d'alerte se présente :

1. Le moniteur d'alertes de réplication lit les critères d'alerte et le contact associé (pour les qualificatifs du moniteur) dans la table IBMSNAP\_CONDITIONS.
2. Pour chaque serveur de contrôle Capture ou Apply possédant un critère d'alerte défini, le moniteur d'alertes de réplication effectue les tâches suivantes :
  - a. Le moniteur d'alertes de réplication se connecte au serveur et lit les tables de contrôle de réplication associées à chaque critère d'alerte afin de vérifier si certains des critères sont réunis.
  - b. Si un critère est atteint, le moniteur d'alertes de réplication conserve en mémoire les données liées à ce critère et poursuit le traitement des autres critères d'alerte pour ce serveur.
  - c. Une fois qu'il a terminé le traitement de tous les critères d'alerte pour ce serveur, le moniteur d'alertes de réplication se déconnecte du serveur de contrôle Capture ou Apply, insère les alertes dans la table IBMSNAP\_ALERTS et notifie les contacts associés à ce critère.

---

## Chapitre 18. Vérification du statut des programmes de réplication SQL

Les rubriques suivantes décrivent les méthodes que vous pouvez utiliser pour vérifier le statut de votre environnement de réplication.

---

### Vérification du statut des programmes de réplication (z/OS, Linux, UNIX, Windows)

Vous pouvez vérifier rapidement le statut en cours des programmes Capture, Apply ou du moniteur d'alertes de réplication.

Utilisez l'une des commandes suivantes pour vérifier le statut des programmes de réplication :

#### Programme Capture

Commande système **asnccmd**, paramètre **status**

#### Programme Apply

Commande système **asnacmd**, paramètre **status**

#### Moniteur d'alertes de réplication

Commande système **asnmcmd**, paramètre **status**

Lorsque vous demandez le statut d'un programme, vous recevez des messages qui décrivent le statut de chaque unité d'exécution associée à ce programme :

- Le programme Capture possède les quatre unités d'exécution suivantes :
  - Unité d'exécution agent
  - Unité d'exécution d'administration
  - Unité d'exécution d'élagage
  - Unité d'exécution de sérialisation
  - Unité d'exécution du programme de lecture des transactions (si le paramètre de démarrage **asynchlogrd** est défini sur oui)
- Le programme Apply possède les unités d'exécution suivantes :
  - Unité d'exécution d'administration (selon votre niveau de maintenance, cela peut ne pas être une unité d'exécution séparée)
  - Unité d'exécution agent
  - Unité d'exécution de sérialisation
  - Unité d'exécution du moniteur (si la colonne **MONITOR\_ENABLED** de la table **IBMSNAP\_APPPARMS** est réglée sur Y)
- Le moniteur d'alertes de réplication possède les trois unités d'exécution suivantes :
  - Unité d'exécution d'administration
  - Unité d'exécution agent
  - Unité d'exécution de sérialisation

Utilisez les messages que vous recevez pour déterminer si vos programmes fonctionnent correctement. En règle générale, les unités d'exécution agent, d'administration et d'élagage sont à l'état opérationnel et effectuent les tâches pour

lesquelles elles ont été conçues. Les unités d'exécution de sérialisation, les gestionnaires de signaux globaux sont généralement en attente de signaux.

L'unité d'exécution d'élagage est chargée d'élaguer les tables de modification des données et les tables de contrôle de réplication suivantes.

- Table IBMSNAP\_UOW
- Table IBMSNAP\_CAPTRACE
- Table IBMSNAP\_CAPMON
- Table IBMSNAP\_SIGNAL

L'unité d'exécution agent prend le statut "attente de la base de données" s'il ne peut pas se connecter à son serveur de contrôle Apply et que le programme Apply a été démarré avec le paramètre **term=n**. Vous pouvez exécuter la commande **asnacmd** ou **MODIFY** sous z/OS pour vérifier si l'unité d'exécution agent est en cours d'exécution mais qu'elle ne peut pas se connecter au serveur de contrôle.

Si les messages que vous recevez indiquent qu'un programme fonctionne, mais que votre environnement indique le contraire, vous devez pousser plus avant vos recherches. Par exemple, si vous vérifiez le statut du programme Apply et qu'il s'avère que l'unité d'exécution agent est en cours de fonctionnement, alors que les données ne sont pas appliquées dans les tables cible comme prévu, examinez la table **IBMSNAP\_APPLYTRAIL** pour rechercher des messages susceptibles d'expliquer pourquoi les données ne sont pas appliquées. Des incidents au niveau des ressources système peuvent empêcher le programme de fonctionner correctement.

---

## Vérification du statut des travaux de journal Capture et Apply (System i)

Sur DB2 pour System i, utilisez la commande système Work with Subsystem Jobs (**WRKSBSJOB**) pour vérifier le statut des travaux de journal pour les programmes Capture et Apply.

### Procédure

Pour vérifier le statut des travaux de journal pour les programmes Capture et Apply :

Utilisez la commande système Work with Subsystem Jobs (**WRKSBSJOB**) comme suit :

1. Saisissez la commande :

```
WRKSBSJOB subsystem
```

Où *subsystem* représente le nom du sous-système. Dans la plupart des cas, le sous-système est QZSNDPR, sauf si vous avez créé votre propre description du sous-système

2. Identifiez les travaux intéressants à partir de ceux répertoriés comme étant en cours d'exécution.

Le travail de journal est nommé en fonction du journal auquel il a été assigné. Si aucun travail n'est répertorié ici, utilisez la commande système Work with Submitted Jobs (**WRKSBMJOB**) ou la commande système Work with Job (**WRKJOB**) pour localiser le travail. Recherchez l'historique relatif au travail afin de vérifier qu'il s'est achevé avec succès ou pour identifier la raison de son échec.

---

## Suivi de l'avancement du programme Capture (System i)

Si le programme Capture est terminé, vous pouvez inspecter la table IBMSNAP\_RESTART pour déterminer la progression réalisée par le programme Capture avant la fin du programme. Il y a une ligne pour chaque journal utilisé par les tables sources. La colonne LOGMARKER fournit l'horodatage de la dernière entrée de journal traitée avec succès. La colonne SEQNBR fournit le numéro de séquence de cette entrée de journal.

### A propos de cette tâche

Si le programme Capture est terminé, vous pouvez inspecter la table IBMSNAP\_RESTART pour déterminer la progression réalisée par le programme Capture avant la fin du programme. Il y a une ligne pour chaque journal utilisé par les tables sources. La colonne LOGMARKER fournit l'horodatage de la dernière entrée de journal traitée avec succès. La colonne SEQNBR fournit le numéro de séquence de cette entrée de journal.

### Procédure

Pour déterminer la progression du programme Capture pendant qu'il est en cours d'exécution :

1. Ouvrez la table de données de modification pour chaque table source capturée.
2. Dans la dernière ligne de chaque table de données de modification, notez la valeur hex dans la colonne COMMITSEQ.
3. Identifiez une ligne dans la table IBMSNAP\_UOW ayant la même valeur hex COMMITSEQ. Si aucune valeur COMMITSEQ correspondante n'existe dans la table IBMSNAP\_UOW, répétez le processus avec l'avant-dernière ligne de la table de données de modification. Avancez à reculons dans la table de données de modification jusqu'à ce que vous identifiez une valeur hex correspondante.
4. Lorsque vous trouvez une valeur hex COMMITSEQ correspondante, notez cette valeur dans la colonne LOGMARKER de la ligne de la table des unités d'oeuvre. Il s'agit de l'horodatage de la dernière entrée de journal traitée. Toutes les modifications de la table source jusqu'à ce moment sont prêtes à être appliquées.
5. Utilisez la commande système Display Journal (**DSPJRN**) pour déterminer combien d'entrées de journal doivent encore être traitées par le programme Capture. Dirigez la sortie dans un fichier de sortie (ou imprimante) pour conserver le rapport, comme indiqué dans l'exemple ci-dessous :

```
DSPJRN FILE(JRNLIB/DJRN1)
      RCVRNG(*CURCHAIN)
      FROMTIME(timestamp)
      TOTIME(*LAST)
      JRNCDE(J F R C)
      OUTPUT(*OUTFILE)
      ENTDTALEN(1) OUTFILE(library/outfile)
```

où *timestamp* est l'horodatage que vous avez identifié à l'étape 4.

Le nombre d'enregistrements dans le fichier de sortie correspond au nombre approximatif d'entrées de journal devant encore être traitées par le programme Capture.





---

## Chapitre 19. Personnalisation et exécution de scripts SQL pour la réplication

Pour créer des tables de contrôle, vous devez enregistrer des tables source et créer des ensembles d'abonnements (avec membres éventuellement) ; vous devez exécuter des scripts SQL générés par le centre de réplication et par le programme ASNCLP. Vous pouvez exécuter les scripts SQL à l'aide du Centre de réplication ou à partir d'une ligne de commande DB2. Le cas échéant, vous pouvez modifier les scripts SQL pour les adapter à vos besoins.

### Avant de commencer

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, puis modifier les instructions SQL pour spécifier l'ID utilisateur et le mot de passe requis sur le serveur auquel vous êtes connecté. Par exemple, recherchez une ligne qui ressemble à l'exemple suivant et ajoutez vos informations en écrasant les paramètres fictifs (XXXX) :

```
CONNECT TO bdsrc USER XXXX USING XXXX ;
```

### A propos de cette tâche

Vous avez la possibilité d'exécuter, dans le programme ASNCLP et dans le Centre de réplication, un script SQL généré soit immédiatement, soit ultérieurement (dans ce cas, vous pouvez l'enregistrer). Même si vous choisissez d'exécuter le script SQL maintenant, vous pouvez l'enregistrer en vue de le consulter ultérieurement. Par exemple, si vous enregistrez les définitions d'un ensemble d'abonnements volumineux dans un fichier SQL, vous pouvez réexécuter les définitions en cas de besoin.

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.
- Définir la taille des espaces table ou des bases de données des tables CD.
- Définir les normes propres au site.
- 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.
- Créer et modifier des instructions de table et d'index pour représenter des objets de base de données.
- Pour Informix et autres bases de données relationnelles non DB2, vérifiez que les tables sont créées dans les espaces de base de données ou dans les espaces table souhaités.

- Pour Microsoft SQL Server, créez des tables de contrôle sur un segment existant.
- Révisez et modifiez les prédicats de membres d'ensembles d'abonnements pour définir simultanément plusieurs ensembles d'abonnements. Vous pouvez utiliser des variables de substitution dans vos prédicats et résoudre les variables avec une logique de programmation.

### **Procédure**

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 nomfichier`
- Utilisez cette commande si le script SQL porte un autre caractère comme délimiteur (dans cet exemple, comme dans la réplication hétérogène, le signe dièse( # ) représente le caractère de fin : `db2 -td# -vf nomfichier`

---

## Chapitre 20. Règles d'attribution de nom applicables aux objets de la réplication SQL

Le tableau ci-dessous répertorie les limites pour les noms d'objets dans la réplication d'objets.

Tableau 20. Limites pour les noms d'objets dans la réplication d'objets

Objet	Limites des noms
Tables source et cible	<p><b>Linux UNIX Windows</b> Respectez les règles d'attribution de nom applicables à votre système de gestion de bases de données.</p> <p><b>System i</b> Les noms ne doivent pas inclure de caractères blancs, d'astérisque (*), de point d'interrogation (?) d'apostrophe ('), de guillemets (") ou de barre oblique (/).</p>
Colonnes source et cible	Respectez les règles d'attribution de nom applicables à votre système de gestion de bases de données. (Vous noterez que toutes les colonnes d'images avant possèdent un préfixe composé d'un caractère.) Pour éviter l'utilisation de noms de colonnes d'images avant ambigus, vérifiez que les noms de colonnes 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.
Ensemble d'abonnements	Un nom d'ensemble d'abonnements peut inclure tout caractère autorisé par DB2 pour les colonnes à caractères variables (VARCHAR). <b>Recommandation :</b> Suivez les règles d'affectation de nom de tables et de colonnes DB2. La réplication DB2 stocke le nom de l'ensemble d'abonnements sur chaque serveur de contrôle de réplication ; par conséquent, vérifiez que ce nom est compatible avec les pages de codes des trois serveurs.
Schéma Capture	Le schéma Capture peut être une chaîne de 128 caractères ou moins <sup>1</sup> . <p><b>z/OS</b> <b>Sous-systèmes utilisant la version 8 en mode compatibilité ou version plus ancienne :</b> 18 caractères maximum</p> <p><b>System i</b> Le schéma Capture (CAPCTLLIB) peut être une chaîne de 10 caractères alphanumériques (ou moins) <sup>1</sup>.</p>
Qualificatif Apply	<p><b>z/OS</b> <b>Linux UNIX Windows</b> Le qualificatif Apply peut être une chaîne de 18 caractères ou moins <sup>1</sup>.</p> <p><b>System i</b> Le qualificatif Apply peut être une chaîne de 18 caractères ou moins ; cependant, les travaux Apply ne pouvant contenir que 10 caractères, les 10 premiers caractères doivent être uniques pour un qualificatif Apply donné <sup>1</sup>.</p>
Qualificatif de moniteur	<p><b>z/OS</b> <b>Linux UNIX Windows</b> Le qualificatif Monitor peut être une chaîne de 18 caractères ou moins <sup>1</sup>.</p>

Tableau 20. Limites pour les noms d'objets dans la réplication d'objets (suite)

Objet	Limites des noms
<b>Remarque :</b>	
1. Pour les schémas Capture, les qualificatifs Apply et les qualificatifs du moniteur, veuillez à n'utiliser que les caractères valides suivants dans les noms de ces objets :	
<ul style="list-style-type: none"><li>• A à Z (lettres majuscules)</li><li>• a à z (lettres minuscules)</li><li>• nombres (0 à 9)</li><li>• Le caractère de soulignement " _ "</li></ul>	
Les blancs ne sont pas autorisés, ni les autres caractères spéciaux, tels que les deux points (":") ou le signe plus ("+").	

Les commandes du système de réplication et du Centre de réplication, par défaut, convertissent en majuscules tous les noms que vous indiquez. 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` or `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** Sous le système d'exploitation Windows, vous devez utiliser un chemin unique pour distinguer les noms qui sont identiques. Par exemple, supposons que vous ayez trois qualificatifs Apply : `myqual`, `MyQual` et `MYQUAL`. Les trois noms utilisent les mêmes caractères mais avec des casses différentes. Si les trois qualificatifs sont dans le même répertoire sur le serveur Apply, ils entraîneront des conflits de noms.

**Important :** Lorsque vous configurez les services Windows pour le programme Capture, le programme Apply ou le moniteur d'alertes de réplication, vous devez utiliser des noms uniques pour le schéma Capture, le schéma Apply et le qualificatif de moniteur. Vous ne pouvez pas vous servir de la casse pour différencier les noms.

---

## Chapitre 21. Commandes système pour la réplication SQL (Linux, UNIX, Windows, z/OS)

Cette section décrit les commandes à utiliser sous Linux, UNIX, Windows et USS (UNIX System Services) sur z/OS pour démarrer, utiliser, modifier et surveiller des programmes de réplication SQL.

Toutes ces commandes sont précédées du préfixe **asn** et sont saisies à l'invite de commande du système d'exploitation ou dans un script d'interpréteur de commande. Une de ces commandes, **asnanalyze**, fonctionne également avec les données distantes qui se trouvent sur les systèmes d'exploitation System i.

---

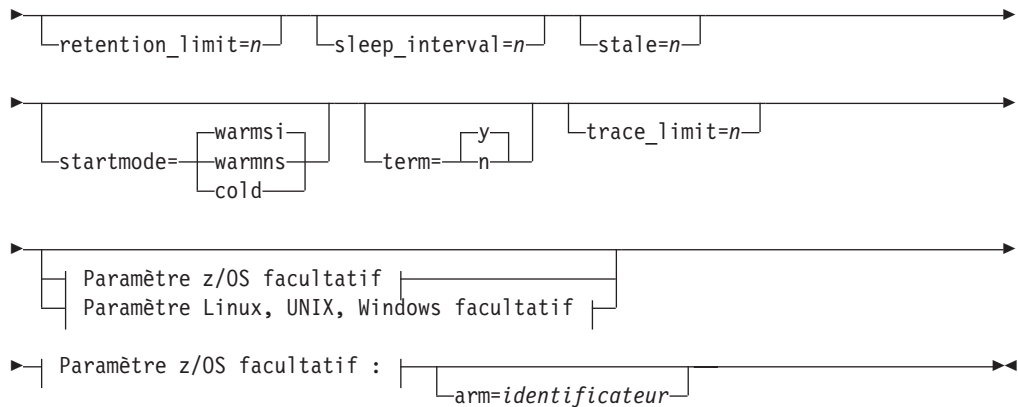
### asncap : démarrage de Capture

Utilisez la commande **asncap** pour démarrer le programme Capture sur Linux, UNIX, Windows et UNIX System Services (USS) sur z/OS. Exécutez cette commande à l'invite du système d'exploitation ou dans un script de shell.

Une fois le programme Capture démarré, il s'exécute en continu jusqu'à ce que vous l'arrêtiez ou qu'il détecte une erreur irréparable.

#### Syntaxe





### Paramètre Linux, UNIX, Windows facultatif :



## Paramètres

Le tableau 21 définit les paramètres d'appel.

Tableau 21. Définitions des paramètres d'appel d'asncap pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS

Paramètre	Définition
<code>capture_server=nom_bd</code>	Indique le nom du serveur de contrôle Capture.  <div style="background-color: #e0e0e0; padding: 5px; margin-bottom: 5px;"><b>z/OS</b> Spécifie le nom du sous-système DB2 sur lequel le programme Capture s'exécutera. Pour le partage de données, n'employez pas le nom de liaison de groupe. A la place, entrez le nom d'un membre du sous-système.</div> <div style="background-color: #e0e0e0; padding: 5px;"><b>Linux UNIX Windows</b> Si vous n'indiquez pas de serveur de contrôle Capture, ce paramètre a par défaut la valeur de la variable d'environnement DB2DBDFT.</div>
<code>add_partition=0/n</code>	<div style="background-color: #e0e0e0; padding: 5px; margin-bottom: 5px;"><b>Linux UNIX Windows</b> Indique si le programme Capture doit commencer à lire le fichier journal des partitions qui ont été ajoutées depuis le dernier redémarrage du programme Capture.</div> <p><b>n (par défaut)</b> Aucune nouvelle partition n'a été ajoutée depuis le dernier redémarrage du programme Capture.</p> <p><b>y</b> Le programme Capture lance la lecture du fichier journal sur une ou plusieurs des nouvelles partitions. Sur chaque partition, le programme Capture lance la lecture du journal à partir d'un numéro d'ordre du journal utilisé au dernier démarrage de la base de données.</p>

Tableau 21. Définitions des paramètres d'appel d'asncap pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>arm</b> = <i>identificateur</i>	<p><b>z/OS</b> Spécifie une chaîne alphanumérique à trois caractères qui est utilisée pour identifier une instance unique du programme Capture sur le Gestionnaire de Redémarrage Automatique. La valeur que vous renseignez est associée au nom d'élément ARM que Capture génère par lui-même : ASNTCxxxxyyyy (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 <b>arm</b>, mais le programme Capture concatènera uniquement trois caractères maximum au nom en cours. Si nécessaire, le programme Capture remplira le nom par des blancs pour composer un nom unique de 16 octets.</p>
<b>asynchlogrd</b> = <i>o/n</i>	<p><b>n (par défaut)</b> Indique le programme Capture doit utiliser la même unité d'exécution pour la lecture du journal de reprise DB2 et le traitement des transactions capturées à partir du journal.</p> <p><b>y</b> Indique le programme Capture doit utiliser la même unité d'exécution dédiée pour capturer les transactions à partir du journal de reprise DB2. L'unité d'exécution du programme de lecture des transactions effectue une prélecture des transactions validées dans une mémoire tampon, d'où une autre unité d'exécution récupère les transactions et les traite en instructions SQL pour les insérer dans la table CD. Ce mode asynchrone permet d'améliorer les performances du programme Capture dans tous les environnements grâce à des avantages particuliers pour les bases de données partitionnées et le partage de données sous z/OS. Sur les systèmes dont les niveaux d'activité sont très élevés, cette prélecture peut entraîner une utilisation plus importante de la mémoire. Ajustez le paramètre <b>memory_limit</b> en conséquence. Si le volume de vos modifications est faible, vous pouvez préférer utiliser la valeur par défaut N pour réduire la consommation d'unité centrale.</p>
<b>capture_schema</b> = <i>schéma</i>	Indique le nom du schéma de Capture servant à identifier un programme Capture déterminé. Le nom du schéma doit comporter de 1 à 128 caractères. Il s'agit par défaut de ASN.
<b>capture_path</b> = <i>chemin</i>	Indique l'emplacement des fichiers de travail utilisés par le programme Capture. Il s'agit par défaut du répertoire où la commande <b>asncap</b> a été appelée.

Tableau 21. Définitions des paramètres d'appel d'asncap pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>autoprune=0/n</b>	<p>Indique si est activée la suppression automatique de lignes dans les tables CD (modification des données), UOW (unités de travail), IBMSNAP_CAPMON, IBMSNAP_CAPTRACE et IBMSNAP_SIGNAL.</p> <p><b>y (par défaut)</b> Le programme Capture supprime automatiquement les lignes admissibles à l'intervalle indiqué dans la table IBMSNAP_CAPPARMS. Il supprime les lignes CD, UOW et IBMSNAP_SIGNAL antérieures à la durée de conservation, qu'elles aient ou non été répliquées.</p> <p><b>n</b> La suppression automatique est désactivée.</p>
<b>autostop=0/n</b>	<p>Indique si le programme Capture se termine après avoir extrait toutes les transactions consignées avant son démarrage.</p> <p><b>n (par défaut)</b> Le programme Capture ne termine pas après l'extraction des transactions.</p> <p><b>y</b> Le programme Capture termine après l'extraction des transactions.</p>
<b>caf=n/o</b>	<p><b>z/OS</b> Le programme Capture s'exécute avec la connexion RRS par défaut (CAF=n). Vous pouvez remplacer cette valeur par défaut et indiquer au programme Capture la fonction de connexion d'appel CAF en définissant l'option <b>caf=y</b>. L'option <b>caf=y</b> indique que le programme de réplication remplace la connexion RRS par défaut et s'exécute avec la connexion CAF.</p> <p><b>n (par défaut)</b> Le programme Capture utilise la connexion RRS (CAF=n).</p> <p><b>y</b> Indique que le programme de réplication remplace la connexion RRS par défaut et s'exécute avec la connexion CAF.</p> <p>Si RRS n'est pas disponible, un message vous l'indique et le programme de réplication 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.</p>
<b>commit_interval=n</b>	<p>Indique le nombre de secondes que le programme Capture attend avant de valider des lignes dans les tables UOW et CD. La valeur par défaut est de 30 secondes.</p>



Tableau 21. Définitions des paramètres d'appel d'asncap pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>ignore_transid=</b> <i>ID_transaction</i>	<p>Indique que le programme Capture ne capturera pas la transaction identifiée sous <i>ID_transaction</i>.</p> <p>La valeur pour <i>ID_transaction</i> est un identificateur hexadécimal de 10 octets au format suivant :</p> <p><b>z/OS</b></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><b>Linux UNIX Windows</b></p> <p><i>mmmm</i>:0000:xxxx:xxxx:xxxx</p> <p>où <i>xxxx:xxxx:xxxx</i> est l'ID transaction et <i>mmmm</i> l'identificateur de partition pour les bases de données partitionnées (0000 pour les bases de données non partitionnées).</p>
<b>lag_limit=</b> <i>n</i>	Indique le nombre de minutes dont le programme Capture dispose pour traiter des enregistrements de journal. La valeur par défaut est de 10080 minutes (sept jours). Le programme Capture vérifie uniquement la valeur de ce paramètre lors d'un démarrage à chaud. Si cette limite est dépassée, le programme Capture ne démarre pas.
<b>logrdbufsz=</b> <i>n</i>	Spécifie la taille de la mémoire tampon que le programme Capture transmet à DB2 lorsque Capture extrait les enregistrements de journaux. DB2 remplit la mémoire tampon avec les enregistrements de journaux disponibles que Capture n'a pas extrait. La valeur par défaut pour DB2 for z/OS est 66 k et 256 Ko pour DB2 for Linux, UNIX et Windows. Pour les bases de données partitionnées, Capture alloue une mémoire tampon de la taille spécifiée par <b>logrdbufsz</b> pour chaque partition.
<b>logread_prefetch=</b> <i>y/n</i>	<p><b>Linux UNIX Windows</b> Indique si le programme Capture recourt à des unités d'exécution séparées pour préextraire les enregistrements de journal de chaque partition dans une base de données partitionnée.</p> <p><b>n (valeur par défaut pour les bases de données non partitionnées)</b></p> <p>Une seule unité d'exécution de lecteur de journal de Capture se connecte à toutes les partitions.</p> <p><b>y (valeur par défaut pour les bases de données partitionnées)</b></p> <p>Une unité d'exécution de lecteur de journal séparée se connecte à chaque partition. L'utilisation d'unités d'exécution séparées permet d'améliorer la capacité de traitement de Capture mais peut accroître l'utilisation de l'unité centrale.</p>

Tableau 21. Définitions des paramètres d'appel d'asncap pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>logreuse=0/n</b>	<p>Définit si le programme Capture réutilise ou ajoute des messages à son fichier journal.</p> <p><b>n (par défaut)</b> Le programme Capture ajoute des messages au fichier journal, même après son redémarrage.</p> <p><b>y</b> Le programme Capture réutilise le fichier journal en tronquant d'abord le premier fichier journal, puis en lançant un nouveau journal à son redémarrage.</p> <p><b>z/OS</b> Le nom du fichier journal ne contient pas le nom d'instance de DB2 : <i>serveur_capture.schéma_capture.CAP.log.</i></p> <p><b>Linux UNIX Windows</b> Le nom du fichier journal comprend le nom d'instance DB2 : <i>instancedb2.serveur_capture.schéma_capture.CAP.log.</i></p>
<b>logstdout=0/n</b>	<p>Indique à quel endroit le programme Capture envoie les messages du fichier journal :</p> <p><b>n (par défaut)</b> Le programme Capture dirige la plupart des messages de fichier journal vers le fichier journal uniquement. Les messages d'initialisation sont envoyés à la fois dans le fichier journal et dans la sortie standard (STDOUT).</p> <p><b>y</b> Le programme Capture envoie les messages du fichier journal à la fois au fichier journal et à la sortie standard (STDOUT).</p>
<b>memory_limit=n</b>	<p>Indique la taille maximum (en mégaoctets) de mémoire que le programme Capture peut utiliser pour générer des transactions. Une fois cette limite atteinte, le programme Capture déverse les transactions dans un fichier. La valeur par défaut est de 32 mégaoctets.</p> <p><b>z/OS</b> Si vous paramétrez <b>memory_limit=0</b>, le programme Capture détermine le volume de mémoire à utiliser à partir du paramètre taille de la région de travail de Capture. L'allocation de mémoire correspond à 80 % de la taille de la région.</p>
<b>migrate=y/n</b>	<p><b>Linux UNIX Windows</b> Spécifie que le programme Capture démarre à partir du début du journal après la migration de DB2.</p> <p><b>Important :</b> N'utilisez cette option que pour le premier démarrage de Capture et spécifiez <b>startmode=warmns</b>. N'utilisez pas non plus le paramètre <b>migrate</b> lorsque vous migrez vers la version 10.1 ou ultérieure.</p>
<b>monitor_interval=n</b>	<p>Indique à quelle fréquence (en secondes) le programme Capture insère des lignes dans la table IBMSNAP_CAPMON. La valeur par défaut est de 300 secondes (cinq minutes).</p>

Tableau 21. Définitions des paramètres d'appel d'asncap pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>monitor_limit</b> = <i>n</i>	Indique combien de temps (en minutes) une ligne peut rester dans la table IBMSNAP_CAPMON avant de pouvoir être supprimée. Toutes les lignes IBMSNAP_CAPMON antérieures à la valeur du paramètre <b>monitor_limit</b> sont supprimées au prochain cycle d'élagage. La valeur par défaut est de 10 080 minutes (sept jours).
<b>part_hist_limit</b> = <i>n</i>	Indique la durée de conservation des anciennes données dans la table IBMQREP_PART_HIST avant leur élagage. La valeur par défaut est de 10080 minutes (sept jours). Ce paramètre contrôle également le point du journal depuis lequel vous pouvez redémarrer le programme Capture, puisque Capture utilise IBMQREP_PART_HIST pour déterminer les archives de journal à lire pour une table source partitionnée.
<b>pwdfile</b> = <i>nomfichier</i>	Spécifie le nom du fichier de mots de passe. Si aucune valeur n'est spécifiée, le fichier asnpwd.aut est utilisé par défaut.  Cette commande recherche le fichier des mots de passe dans le répertoire défini par le paramètre <b>capture_path</b> . Si aucun paramètre <b>capture_path</b> n'est défini, cette commande recherche le fichier des mots de passe dans le répertoire où la commande a été appelée.
<b>prune_interval</b> = <i>n</i>	Indique à quelle fréquence (en secondes) les tables CD, UOW, IBMSNAP_CAPMON, IBMSNAP_CAPTRACE et IBMSNAP_SIGNAL sont supprimées. Ce paramètre est ignoré si vous prenez <i>n</i> comme valeur pour le paramètre autopruner. La valeur par défaut est de 300 secondes (cinq minutes).
<b>prunemsg</b> = <i>y/n</i>	Indique si le programme Capture délivre des messages d'information sur le statut de l'élagage.  <b>y (par défaut)</b> Capture délivre des messages d'information sur le statut de l'élagage.  <b>n</b> Capture ne délivre pas de messages d'information sur le statut de l'élagage.
<b>retention_limit</b> = <i>n</i>	Indique combien de temps (en minutes) une ligne peut rester dans la table CD, UOW ou IBMSNAP_SIGNAL avant de pouvoir être supprimée. Chaque ligne antérieure à la valeur du paramètre <b>retention_limit</b> est supprimée au prochain cycle d'élagage. La valeur par défaut est de 10 080 minutes (sept jours).
<b>sleep_interval</b> = <i>n</i>	Indique le nombre de secondes que le programme Capture se met en veille après avoir traité le journal actif et détecté que la mémoire tampon est vide. La valeur par défaut est de cinq secondes.  <b>z/OS</b> Indique le nombre de secondes que le programme Capture se met en veille si la mémoire tampon est remplie à moins de la moitié.

Tableau 21. Définitions des paramètres d'appel d'asncap pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>stale=<i>n</i></b>	Indique le nombre de secondes pendant lesquelles le programme Capture attend de pouvoir émettre un message d'avertissement ou d'effectuer une autre action après avoir détecté une transaction à exécution longue sans enregistrement de validation ou d'annulation. Le comportement du programme dépend de la plateforme de la source. Sous z/OS, Capture émet des messages d'avertissement s'il ne trouve pas d'enregistrement de validation ou d'annulation pendant une heure ( <b>stale=3600</b> ). Sous z/OS et Linux, UNIX et Windows, si une transaction est exécutée pendant le nombre de secondes indiqué par le paramètre <b>stale</b> et que le programme Capture ne trouve aucune opération de ligne dans le journal pour cette transaction, il émet des messages d'avertissement, ne réplique pas la transaction et passe au numéro de séquence 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.
<b>startmode=<i>mode</i></b>	Indique la procédure de traitement utilisée par le programme Capture au démarrage. <p><b>warmsi (par défaut)</b></p> <p>Le programme Capture reprend le traitement là où il s'était arrêté à son exécution antérieure si des informations sur le démarrage à chaud sont disponibles. S'il s'agit du premier démarrage du programme Capture, il effectue automatiquement un démarrage à froid.</p> <p>Lors d'un démarrage à chaud, le programme Capture garde les tables IBMSNAP_CAPTRACE, CD, UOW et IBMSNAP_RESTART inchangées. Si des erreurs se produisent après le démarrage du programme Capture, celui-ci se termine.</p> <p><b>warmns</b></p> <p>Le programme Capture reprend le traitement là où il s'était arrêté à son exécution antérieure si des informations sur le démarrage à chaud sont disponibles. Si des erreurs se produisent après le démarrage du programme Capture, celui-ci se termine. S'il ne peut pas être démarré à chaud, il ne passe pas à un démarrage à froid.</p> <p><b>cold</b></p> <p>Le programme Capture commence à supprimer toutes les lignes dans ses tables CD et UOW. La plupart des enregistrements sont réinitialisés afin que tous les abonnements à ces sources soient totalement régénérés au cours du prochain cycle de traitement d'Apply. Les enregistrements pour des tables CCD externes et les abonnements dont les cibles sont des tables CCD incomplètes ne sont pas totalement régénérés.</p>

Tableau 21. Définitions des paramètres d'appel d'asncap pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>term=0/n</b>	<p>Indique si le programme Capture se termine en cas d'arrêt ou de mise au repos de DB2.</p> <p><b>y (par défaut)</b> Le programme Capture se termine en cas d'arrêt ou de mise au repos de DB2.</p> <p><b>n</b> Le programme Capture continue de fonctionner en cas d'arrêt ou de mise au repos de DB2. A l'initialisation de DB2, le programme Capture démarre la capture au point où il s'était arrêté à la fermeture ou à la mise en repos de DB2.</p> <p>Si DB2 se termine via FORCE ou de manière anormale, le programme Capture se termine aussi, même si ce paramètre est réglé sur n.</p> <p>Si vous définissez ce paramètre sur n et démarrez DB2 avec un accès restreint (ACCESS MAINT), le programme Capture ne peut pas se connecter et se termine.</p>
<b>trace_limit=n</b>	<p>Indique combien de temps (en minutes) une ligne peut rester dans la table IBMSNAP_CAPTRACE avant de pouvoir être supprimée. Toutes les lignes IBMSNAP_CAPTRACE antérieures à la valeur du paramètre <b>trace_limit</b> sont supprimées au prochain cycle d'élagage. La valeur par défaut est de 10 080 minutes (sept jours).</p>

## Codes retour

La commande **asncap** renvoie un code de retour zéro si elle aboutit. Un code retour autre que zéro est renvoyé si la commande ne s'est pas exécutée normalement.

## Exemples pour asncap

Les exemples suivants illustrent comment utiliser la commande **asncap**.

### Exemple 1

Pour le premier démarrage d'un programme Capture avec un serveur de contrôle Capture nommé db et un schéma de Capture ASN dont les fichiers de travail se trouvent dans le répertoire /home/files/capture/logs/ :

```
asncap capture_server=db capture_schema=ASN
capture_path=/home/files/capture/logs/ startmode=cold
```

### Exemple 2

Pour redémarrer un programme Capture sans suppression une fois celui-ci arrêté :

```
asncap capture_server=db autoprunce=n sleep_interval=10 startmode=warmsi
```

Dans cet exemple, le programme Capture conserve toutes les lignes dans les tables de contrôle correspondantes et se met en veille pendant dix secondes au terme du traitement du journal actif et après avoir détecté que la mémoire tampon est vide.

Le programme Capture reprend le traitement là où il s'était arrêté et passe à un démarrage à froid si les informations sur le démarrage à chaud ne sont pas disponibles.

### Exemple 3

Pour redémarrer un programme Capture en mode démarrage à chaud et modifier les paramètres :

```
asncap capture_server=db autoprun=y prune_interval=60 retention_limit=1440
      startmode=warms
```

Cette commande redémarre le programme Capture et utilise de nouveaux paramètres pour que les tables CD, UOW et IBMSNAP\_SIGNAL puissent plus rapidement être supprimées, ainsi que pour augmenter la fréquence de suppression des paramètres par défaut. Le programme Capture reprend le traitement là où il s'était arrêté à son exécution antérieure mais ne passe pas forcément à un démarrage à froid si les informations sur le démarrage à chaud ne sont pas disponibles.

### Exemple 4

Pour démarrer un programme Capture envoyant tous ses fichiers de travail à un nouveau sous-répertoire capture\_files :

1. Allez au répertoire approprié et créez un sous-répertoire capture\_files :

```
cd /home/db2inst
mkdir capture_files
```

2. Démarrez le programme Capture et indiquez un chemin d'accès figurant dans le sous-répertoire créé :

```
asncap capture_server=db capture_schema=ASN
      capture_path=/home/db2inst/capture_files startmode=warmsi
```

---

## asncmd : fonctionnement de Capture

Utilisez la commande **asncmd** pour envoyer une commande à un programme Capture en cours d'exécution sur Linux, UNIX, Windows et UNIX System Services (USS) sur z/OS. Exécutez cette commande à l'invite du système d'exploitation ou dans un script de shell.

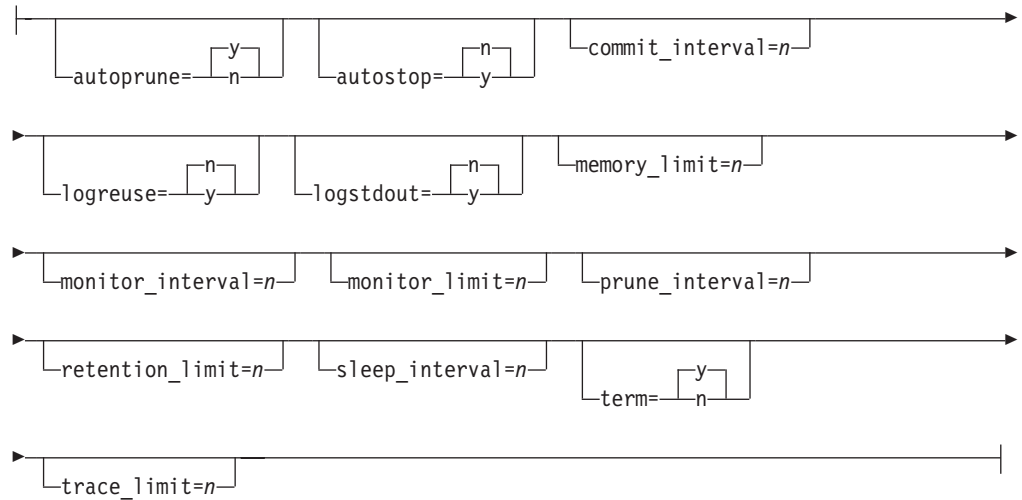
### Syntaxe

Pour plus d'informations sur l'utilisation de la commande MVS MODIFY pour envoyer des commandes à un programme Capture en cours d'exécution sur z/OS, voir Utilisation des programmes de réplication SQL en cours d'exécution à l'aide de la commande MVS MODIFY.

```
▶▶ asncmd [capture_server=nom_bd] [capture_schema=schéma]
```



### Paramètres :



### Paramètres

Le tableau 22 définit les paramètres d'appel pour la commande asncmd.

Tableau 22. Définitions des paramètres d'appel asncmd

Paramètre	Définition
<code>capture_server=o/n</code>	Indique le nom du serveur de contrôle Capture.  <div style="background-color: #800000; color: white; padding: 2px; text-align: center; font-weight: bold;">z/OS</div> <p>Nom du serveur de base de données se connectant au serveur de contrôle. Pour le partage de données, prenez le nom de liaison de groupe ou le nom d'un membre du sous-système.</p> <div style="background-color: #800000; color: white; padding: 2px; text-align: center; font-weight: bold;">Linux UNIX Windows</div> <p>Si vous n'indiquez pas de serveur de contrôle Capture, ce paramètre a par défaut la valeur de la variable d'environnement <code>DB2DBDFT</code>.</p>
<code>capture_schema=schéma</code>	Indique le nom du schéma de Capture servant à identifier un programme Capture déterminé. Le nom du schéma doit comporter de 1 à 128 caractères. Il s'agit par défaut de <code>ASN</code> .

Tableau 22. Définitions des paramètres d'appel `asncmd` (suite)

Paramètre	Définition
<b>chgparms</b>	<p>Indique qu'il faut modifier un ou plusieurs des paramètres d'exploitation suivants d'un programme Capture pendant son exécution :</p> <ul style="list-style-type: none"> <li>• <b>autostop</b></li> <li>• <b>commit_interval</b></li> <li>• <b>logreuse</b></li> <li>• <b>logstdout</b></li> <li>• <b>memory_limit</b></li> <li>• <b>monitor_interval</b></li> <li>• <b>monitor_limit</b></li> <li>• <b>prune_interval</b></li> <li>• <b>retention_limit</b></li> <li>• <b>signal_limit</b></li> <li>• <b>sleep_interval</b></li> <li>• <b>term</b></li> <li>• <b>trace_limit</b></li> </ul> <p><b>Restriction :</b> <span style="background-color: #d9534f; color: white; padding: 2px;">z/OS</span> La valeur de <b>memory_limit</b> n'est pas modifiable si le programme Capture est en cours d'exécution. Pour modifier cette valeur, vous devez d'abord arrêter le programme Capture.</p> <p>Vous pouvez indiquer plusieurs paramètres dans une commande <b>asncmd chgparms</b> et modifier leur valeur aussi souvent que vous le souhaitez. Les modifications écrasent temporairement les valeurs dans la table <code>IBMSNAP_CAPPARMS</code>, mais elles ne sont pas intégrées à la table. Lorsque vous arrêtez et redémarrez le programme Capture, celui-ci utilise les valeurs dans <code>IBMSNAP_CAPPARMS</code>. «<code>asncap</code> : démarrage de Capture», à la page 257 contient la description des paramètres que vous pouvez remplacer à l'aide de cette commande.</p>
<b>prune</b>	<p>Entrez ce paramètre si vous voulez supprimer les tables <code>CD</code>, <code>UOW</code>, <code>IBMSNAP_CAPMON</code>, <code>IBMSNAP_CAPTRACE</code> et <code>IBMSNAP_SIGNAL</code>. Le programme Capture émet un message lorsque la mise en file d'attente de la commande aboutit.</p>
<b>qryparms</b>	<p>Indiquez si les valeurs du paramètre opérationnel en cours doivent être écrites dans la sortie standard (<code>stdout</code>).</p>
<b>reinit</b>	<p>Entrez ce paramètre pour que le programme Capture obtienne les sources de réplication ajoutées de la table <code>IBMSNAP_REGISTER</code>. Par exemple, utilisez ce paramètre si vous ajoutez une source de réplication et si vous employez l'instruction <code>ALTER ADD</code> pour ajouter une colonne à une source de réplication et à la table <code>CD</code> pendant l'exécution du programme Capture.</p>



Tableau 22. Définitions des paramètres d'appel `asnccmd` (suite)

Paramètre	Définition
<b>suspend</b>	Entrez ce paramètre pour libérer des ressources du système d'exploitation pour des transactions opérationnelles à des heures pleines sans détruire l'environnement du programme Capture.  <b>Avertissement :</b> N'interrompez pas Capture pour annuler une source de réplication. A la place, arrêtez le programme.
<b>resume</b>	Entrez ce paramètre pour que le programme Capture interrompu reprenne la capture de données.
<b>status</b>	Entrez ce paramètre pour recevoir des messages indiquant l'état de chaque unité d'exécution de Capture (administration, élagage, sérialisation et tâche).
<b>stop</b>	Entrez ce paramètre pour arrêter correctement le programme Capture et valider les enregistrements de journal traités jusqu'à ce point.

## Exemples pour `asnccmd`

Les exemples suivants illustrent comment utiliser la commande `asnccmd`.

### Exemple 1

Pour permettre à un programme Capture en cours d'exécution d'identifier des sources de réplication ajoutées :

```
asnccmd capture_server=db capture_schema=ASN reinit
```

### Exemple 2

Pour supprimer une fois les tables CD, UOW, IBMSNAP\_CAPMON, IBMSNAP\_CAPTRACE et IBMSNAP\_SIGNAL :

```
asnccmd capture_server=db capture_schema=ASN prune
```

### Exemple 3

Pour recevoir des messages sur l'état de chaque unité d'exécution de Capture :

```
asnccmd capture_server=db capture_schema=ASN status
```

### Exemple 4

Pour envoyer les valeurs opérationnelles en cours d'un programme Capture vers la sortie standard :

```
asnccmd capture_server=db capture_schema=ASN qryparms
```

### Exemple 5

Pour désactiver l'élagage automatique dans un programme Capture en cours d'exécution :

```
asnccmd capture_server=db capture_schema=ASN chgparms autoprune=n
```

### Exemple 6

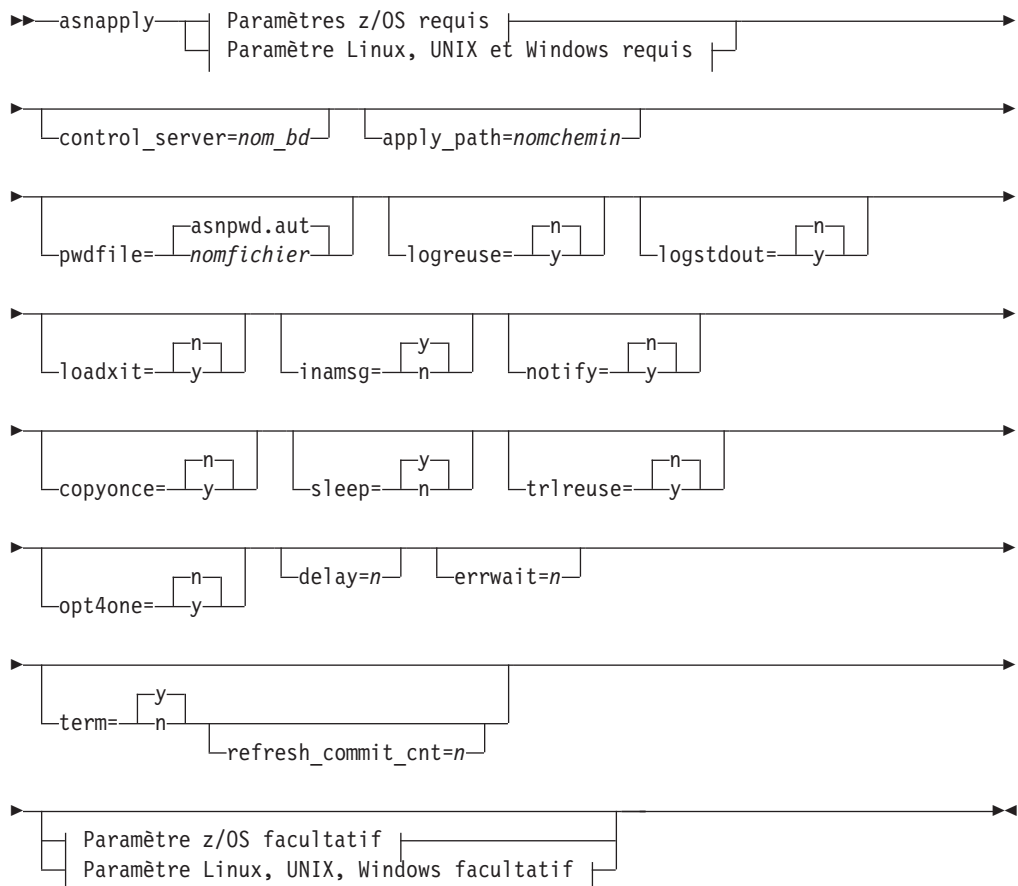
Pour arrêter un programme Capture en cours d'exécution :  
 asncmd capture\_server=db capture\_schema=ASN stop

## asnapply : démarrage du programme Apply

Utilisez la commande **asnapply** pour démarrer le programme Apply sous Linux, UNIX, Windows et UNIX System Services (USS) sur z/OS. Exécutez cette commande à l'invite du système d'exploitation ou dans un script de shell.

Une fois le programme Apply démarré, il s'exécute en continu jusqu'à ce que vous l'arrêtiez ou qu'il détecte une erreur irréparable.

### Syntaxe



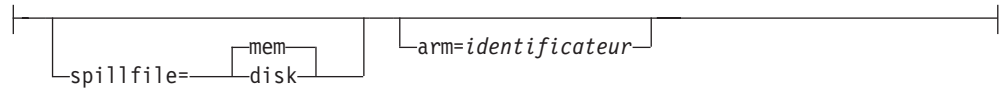
### Paramètres z/OS requis :

`—apply_qual=qual_apply—db2_subsystem=nom—`

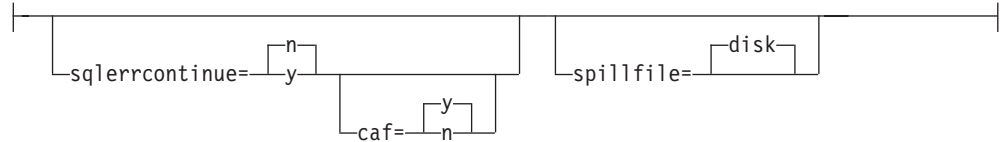
### Paramètre Linux, UNIX, Windows requis :

`—apply_qual=qual_apply—`

### Paramètre z/OS facultatif :



### Paramètre Linux, UNIX, Windows facultatif :



## Paramètres

Le tableau 23 définit les paramètres d'appel.

Tableau 23. Définitions des paramètres d'appel *asnapply* pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS

Paramètre	Définition
<code>apply_qual=qual_apply</code>	Indique le qualificatif Apply que le programme Apply utilise pour identifier les ensembles d'abonnements à prendre en charge.  La valeur que vous saisissez doit correspondre à la valeur de la colonne APPLY_QUAL dans la table IBMSNAP_SUBS_SET. Le nom de qualificatif Apply est sensible à la casse et peut comporter 18 caractères au maximum.
<code>db2_subsystem=nom</code>	<b>z/OS</b> Spécifie le nom du sous-système DB2 sur lequel le programme Apply s'exécutera. Le nom de sous-système entré peut contenir au maximum quatre caractères. Il n'existe pas de valeur par défaut pour ce paramètre. Ce paramètre est obligatoire.
<code>control_server=nom_bd</code>	Spécifie le nom du serveur de contrôle Apply sur lequel les définitions d'abonnements et les tables de contrôle du programme Apply résident.  <b>z/OS</b> Indique le nom du serveur de contrôle de Apply.  <b>Linux UNIX Windows</b> Si vous n'indiquez pas de serveur de contrôle de Apply, ce paramètre a par défaut la valeur de la variable d'environnement DB2DBDFT.
<code>apply_path=nomchemin</code>	Spécifie l'emplacement des fichiers de travail utilisés par le programme Apply. Par défaut, il s'agit du répertoire dans lequel la commande <b>asnapply</b> a été appelée.

Tableau 23. Définitions des paramètres d'appel asnaply pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>pwdfile</b> = <i>nomfichier</i>	<p>Spécifie le nom du fichier de mots de passe. Si aucune valeur n'est spécifiée, le fichier asnpwd.aut est utilisé par défaut.</p> <p>Cette commande recherche le fichier de mots de passe dans le répertoire spécifié par le paramètre <b>apply_path</b>. Si aucun paramètre <b>apply_path</b> n'est spécifié, cette commande recherche le fichier de mots de passe dans le répertoire où la commande a été appelée.</p>
<b>logreuse</b> = <i>o/n</i>	<p>Définit si le programme Apply réutilise ou ajoute des messages à son fichier journal.</p> <p><b>n (par défaut)</b> Le programme Apply ajoute des messages dans le fichier journal, même après que le programme Apply a été redémarré.</p> <p><b>y</b> Le programme Apply réutilise le fichier journal en le supprimant, puis en le créant lorsque le programme Apply est redémarré.</p> <p><b>z/OS</b> Le nom du fichier journal ne contient pas le nom d'instance de DB2 : <i>serveur_contrôle.qual_apply.APP.log</i>.</p> <p><b>Linux UNIX Windows</b> Le nom du fichier journal comprend le nom d'instance de DB2 : <i>instancedb2.serveur_contrôle.qual_apply.APP.log</i></p>
<b>logstdout</b> = <i>o/n</i>	<p>Indique où le programme Apply envoie les messages du fichier journal :</p> <p><b>n (par défaut)</b> Le programme Apply envoie la plupart des messages de fichier journal dans le fichier journal uniquement. Les messages d'initialisation sont envoyés à la fois dans le fichier journal et dans la sortie standard (STDOUT).</p> <p><b>y</b> Le programme Apply envoie les messages de fichier journal à la fois dans le fichier journal et dans la sortie standard (STDOUT).</p>
<b>loadxit</b> = <i>o/n</i>	<p>Indique si le programme Apply appelle ASNLOAD. ASNLOAD est une routine d'exit fournie par IBM, qui passe par les utilitaires d'exportation et de chargement pour actualiser les tables cible.</p> <p><b>n (par défaut)</b> Le programme Apply n'appelle pas ASNLOAD.</p> <p><b>y</b> Le programme Apply appelle ASNLOAD.</p>

Tableau 23. Définitions des paramètres d'appel asnaply pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>inamsg=0/n</b>	Indique si le programme Apply émet un message lorsqu'il est inactif.  <b>y (par défaut)</b> Le programme Apply émet un message lorsqu'il est inactif.  <b>n</b> Le programme Apply n'émet pas de message lorsqu'il est inactif.
<b>notify=0/n</b>	Indique si le programme Apply doit appeler ASNDONE. ASNDONE est une routine d'exit qui vous redonne le contrôle lorsque le programme Apply a terminé de copier un ensemble d'abonnements.  <b>n (par défaut)</b> Le programme Apply n'appelle pas ASNDONE.  <b>y</b> Le programme Apply appelle ASNDONE.
<b>copyonce=0/n</b>	Indique si le programme Apply exécute un cycle de copie pour chaque ensemble d'abonnements éligible au moment où le programme Apply est appelé. Ensuite, le programme Apply s'arrête. Un ensemble d'abonnements éligible remplit les critères suivants : <ul style="list-style-type: none"> <li>• (ACTIVATE &gt; 0) dans la table IBMSNAP_SUBS_SET. Lorsque la valeur de la colonne ACTIVATE est supérieure à zéro, l'ensemble d'abonnements est actif de manière permanente ou est utilisé pour un processus d'abonnement unique.</li> <li>• (REFRESH_TYPE = R ou B) ou (REFRESH_TYPE = E et l'événement spécifié s'est produit). La valeur de la colonne REFRESH_TYPE est stockée dans la table IBMSNAP_SUBS_SET.</li> </ul> <p>La limite MAX_SYNCH_MINUTES de la table des ensembles d'abonnements et l'horodatage END_OF_PERIOD de la table IBMSNAP_SUBS_EVENT sont pris en compte s'ils sont spécifiés.</p> <b>n (par défaut)</b> Le programme Apply n'exécute pas un cycle de copie pour chaque ensemble d'abonnements éligible.  <b>y</b> Le programme Apply exécute un cycle de copie pour chaque ensemble d'abonnements éligible.
<b>sleep=0/n</b>	Indique comment le programme Apply doit se comporter si aucun autre ensemble d'abonnements n'est éligible pour traitement.  <b>y (par défaut)</b> Le programme Apply se met en veille.  <b>n</b> Le programme Apply est arrêté.

Tableau 23. Définitions des paramètres d'appel *asnapply* pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>tr1reuse</b> = <i>o/n</i>	<p>Indique si le programme Apply vide la table IBMSNAP_APPLYTRAIL lorsqu'il démarre.</p> <p><b>n (par défaut)</b> Le programme Apply ajoute des entrées à la table IBMSNAP_APPLYTRAIL. Le programme Apply ne vide pas la table.</p> <p><b>y</b> Le programme Apply vide la table IBMSNAP_APPLYTRAIL lorsqu'il démarre.</p>
<b>opt4one</b> = <i>o/n</i>	<p>Indique si les performances du programme Apply sont optimisées si un seul ensemble d'abonnements est défini pour le programme Apply.</p> <p><b>n (par défaut)</b> Les performances du programme Apply ne sont pas optimisées pour un ensemble d'abonnements.</p> <p><b>y</b> Les performances du programme Apply sont optimisées pour un ensemble d'abonnements. Si vous avez défini la valeur <i>y</i>, le programme Apply met en cache et réutilise les informations concernant les membres des ensembles d'abonnements. Cela permet de réduire l'utilisation de l'UC et d'améliorer le débit.</p>
<b>delay</b> = <i>n</i>	<p>Indique le délai (en secondes) à la fin de chaque cycle Apply lorsque la réplication continue est utilisée, avec <i>n</i>=0, 1, 2, 3, 4, 5 ou 6. La valeur par défaut est 6 et elle est utilisée en cas de réplication continue (c'est-à-dire lorsque l'ensemble d'abonnements utilise le paramètre <i>sleep</i>=0 minutes). Ce paramètre est ignoré si vous avez spécifié <b>copyonce</b>.</p>
<b>errwait</b> = <i>n</i>	<p>Indique le délai en secondes (1 à 65535) que le programme Apply laisse passer avant de faire une nouvelle tentative, lorsqu'il rencontre une condition d'erreur. La valeur par défaut est 300 secondes (cinq minutes).</p> <p><b>Remarque :</b> Ne choisissez pas une valeur trop petite, car le programme Apply s'exécute de manière quasi-continue et génère de nombreuses lignes dans la table IBMSNAP_APPLYTRAIL.</p>
<b>term</b> = <i>o/n</i>	<p>Indique si le programme Apply continue à s'exécuter s'il ne peut pas se connecter à son serveur de contrôle.</p> <p><b>y (par défaut)</b> Par défaut, le programme Apply s'arrête s'il ne peut pas se connecter à son serveur de contrôle.</p> <p><b>n</b> Le programme Apply ne s'arrête pas. Il consigne une erreur, attend le temps défini par le paramètre <b>errwait</b> et réessaie de se connecter.</p> <p>Ce paramètre est ignoré si vous avez spécifié <b>copyonce</b>.</p>

Tableau 23. Définitions des paramètres d'appel asnaply pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>spillfile</b> =type_fichier	Indique si l'ensemble de réponses extrait est stocké.  <div style="background-color: #c00000; color: white; padding: 2px; display: inline-block;">z/OS</div> Les valeurs valides sont :  <b>mem (par défaut)</b> Un fichier mémoire. Si la quantité de mémoire est insuffisante pour l'ensemble de réponses, le programme Apply utilise un fichier disque.  <b>disk</b> Un fichier disque.  <b>hs</b> Espace de données hautes performances (hiperspace)  <div style="background-color: #c00000; color: white; padding: 2px; display: inline-block;">Linux UNIX Windows</div> Les valeurs valides sont :  <b>disk (par défaut)</b> Un fichier disque.
<b>arm</b> =identificateur	  <div style="background-color: #c00000; color: white; padding: 2px; display: inline-block;">z/OS</div> Spécifie une chaîne alphanumérique à trois caractères qui est utilisée pour identifier une instance unique du programme Apply sur le Gestionnaire de Redémarrage Automatique. La valeur que vous renseignez est associée au nom d'élément ARM que Apply génère par lui-même : ASNTAxxxxyyyy (où xxx 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 <b>arm</b> , mais le programme Apply concaténera uniquement trois caractères maximum au nom en cours. Si nécessaire, le programme Apply remplira le nom par des blancs pour composer un nom unique de 16 octets.
<b>caf</b> =o/n	  <div style="background-color: #c00000; color: white; padding: 2px; display: inline-block;">z/OS</div> Indique si le programme Apply s'exécute avec une connexion RRS (Recoverable Resource Manager Services) (CAF=n). L'option de paramètre d'exécution de fonction de connexion d'appel CAF <b>caf</b> =y indique si le programme de réplication est prioritaire sur la connexion RRS et s'exécute avec la connexion CAF. L'option <b>caf</b> =y est l'option par défaut du programme Apply.  <b>y (par défaut)</b> Indique que le programme Apply s'exécute avec la connexion CAF.  <b>n</b> Le programme Apply utilise la connexion RRS ( <b>caf</b> =n).

Tableau 23. Définitions des paramètres d'appel `asnapply` pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<code>sqlerrcontinue=0/n</code>	<p>Indique si le programme Apply continue de s'exécuter lorsqu'il rencontre certaines erreurs SQL.</p> <p>Le programme Apply vérifie l'instruction <code>SQLSTATE</code> erronée par rapport aux valeurs spécifiées dans le fichier <code>SQLSTATE</code>, que vous créez avant d'exécuter le programme Apply. Si une correspondance est trouvée, le programme Apply écrit les informations concernant la ligne défectueuse dans un fichier d'erreur (<code>apply_qualifier.ERR</code>) et poursuit le traitement. Le fichier <code>SQLSTATE</code> peut contenir jusqu'à 20 valeurs à cinq octets.</p> <p><b>n (par défaut)</b> Le programme Apply ne consulte pas le fichier <code>SQLSTATE</code>.</p> <p><b>y</b> Le programme Apply consulte le fichier <code>SQLSTATE</code> pendant le traitement.</p>
<code>refresh_commit_cnt=n</code>	<p>Au cours d'une régénération intégrale, Apply émet une instruction <code>COMMIT</code> après l'insertion du nombre de lignes spécifié dans la table cible. Les valeurs peuvent être comprises entre 0 et 134217727. Avec une valeur par défaut égale à 0, une seule validation est réalisée après l'insertion de toutes les lignes ; aucune validation intermédiaire n'a lieu.</p>

## Codes retour

La commande `asnapply` renvoie un code retour zéro lorsque l'exécution est terminée. Un code retour autre que zéro est renvoyé si la commande ne s'est pas exécutée normalement.

## Exemples pour `asnapply`

Les exemples suivants illustrent l'utilisation de la commande `asnapply`.

### Exemple 1

Pour démarrer un programme Apply à l'aide d'un qualificatif Apply appelé `AQ1`, un serveur de contrôle appelé `dbx` avec des fichiers de travail situés dans le répertoire `/home/files/apply/` :

```
asnapply apply_qual=AQ1 control_server=dbx apply_path=/home/files/apply/
pwdfile=pass1.txt
```

le programme Apply recherche dans le répertoire `/home/files/apply/` le fichier de mots de passe appelé `pass1.txt`.

### Exemple 2

Pour démarrer un programme Apply qui appelle la routine d'exit `ASNLOAD` :

```
asnapply apply_qual=AQ1 control_server=dbx pwdfile=pass1.txt loadxit=y
```

Dans cet exemple, le programme Apply recherche dans le répertoire actuel le fichier de mots de passe appelé `pass1.txt`.



### Exemple 3

Pour démarrer un programme Apply qui exécute un cycle de copie pour chaque ensemble d'abonnements éligible :

```
asnapply apply_qual=AQ1 control_server=dbx apply_path=/home/files/apply/  
copyonce=y
```

Dans cet exemple, le programme Apply recherche dans le répertoire /home/files/apply/ le fichier de mots de passe par défaut appelé asnpwd.aut.

---

## asnacmd : Apply en fonctionnement

Utilisez la commande **asnacmd** pour faire fonctionner le programme Apply 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 Apply en cours d'exécution sur z/OS, voir Utilisation des programmes de réplication SQL en cours d'exécution à l'aide de la commande MVS MODIFY.

### Syntaxe

```
▶▶ asnacmd—apply_qual=qual_apply—┐──┬── status ──▶  
└── control_server=nom_bd ┘└── stop ┘
```

### Paramètres

Le tableau 24 définit les paramètres d'appel.

*Tableau 24. Définitions des paramètres d'appel asnacmd pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS.*

Paramètre	Définition
<b>apply_qual</b> = <i>qual_apply</i>	Indique le qualificatif Apply que le programme Apply utilise pour identifier les ensembles d'abonnements à prendre en charge.  Vous devez indiquer un qualificatif Apply. La valeur que vous saisissez doit correspondre à la valeur de la colonne APPLY_QUAL dans la table IBMSNAP_SUBS_SET. Le nom de qualificatif Apply est sensible à la casse et peut comporter 18 caractères au maximum.
<b>control_server</b> = <i>nom_bd</i>	Indique le nom du serveur de contrôle Apply sur lequel se trouvent les définitions d'abonnement et les tables de contrôle Apply.  <b>z/OS</b> Le paramètre de serveur de contrôle est le nom du serveur de la base que se connecte au serveur de contrôle.  <b>Linux UNIX Windows</b> Si vous n'indiquez pas de serveur de contrôle de Apply, ce paramètre a par défaut la valeur de la variable d'environnement DB2DBDFT.
<b>status</b>	Indique de recevoir les messages donnant l'état de chaque unité d'exécution (administration et worker) dans Apply.

Tableau 24. Définitions des paramètres d'appel `asnacmd` pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS. (suite)

Paramètre	Définition
<code>stop</code>	Indiquez si vous souhaitez arrêter le programme Apply de manière méthodique.

## Exemples de `asnacmd`

Les exemples suivants illustrent la méthode d'utilisation de la commande `asnacmd`.

### Exemple 1

Pour recevoir des messages relatifs à l'état de chaque unité d'exécution Apply :

```
asnacmd apply_qual=AQ1 control_server=dbx status
```

### Exemple 2

Pour arrêter le programme Apply :

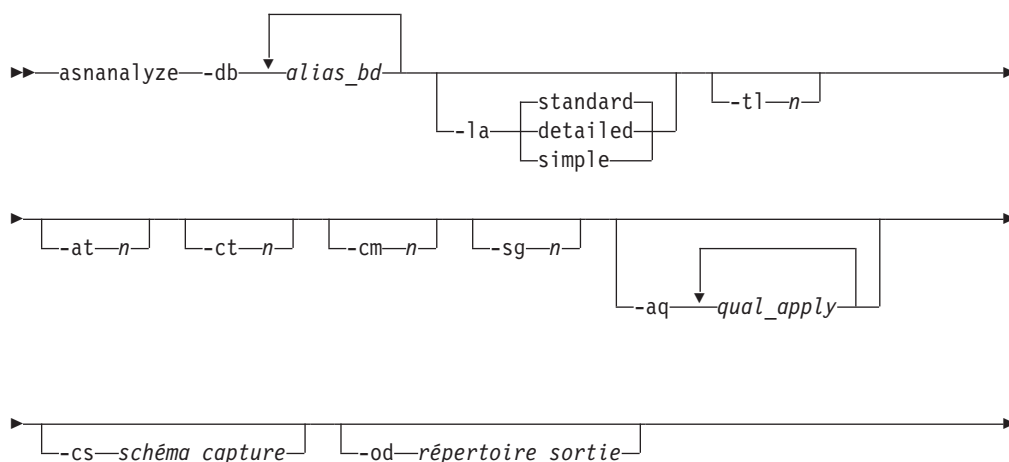
```
asnacmd apply_qual=AQ1 control_server=dbx stop
```

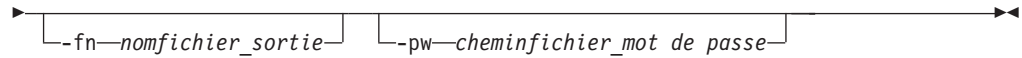
## `asnanalyze` : fonctionnement de l'analyseur

Utilisez la commande `asnanalyze` pour générer des rapports sur l'état des tables de contrôle de réplication. Cette commande analyse les tables de contrôle de réplication qui résident sur tout système d'exploitation, y compris System i ; cependant, vous devez appeler la commande depuis Linux, UNIX or Windows.

Vous devez entrer un espace entre la commande `asnanalyze` et le premier paramètre pour appeler la commande. Si vous émettez la commande sans paramètre, l'aide correspondante s'affiche à l'écran.

### Syntaxe





## Paramètres

Le tableau 25 définit les paramètres d'appel.

Tableau 25. Définitions des paramètres d'appel d'asnanalyze pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Définition
<b>-db</b> <i>alias_bd</i>	Désigne le serveur de contrôle Capture, le serveur cible et le serveur de contrôle Apply.  Vous devez indiquer au moins un alias de base de données. S'il existe plusieurs alias de base de données, séparez les valeurs par des espaces.
<b>-la</b> <i>niveau_analyse</i>	Indique le niveau d'analyse à signaler :  <b>standard (par défaut)</b> Génère un rapport incluant le contenu des tables de contrôle et les informations de statut des programmes Capture et Apply.  <b>detailed</b> Génère les informations dans le rapport standard et : <ul style="list-style-type: none"> <li>• des informations sur les tables CD et des unités de travail,</li> <li>• des informations sur le partitionnement et la compression de l'espace table DB2 z/OS,</li> <li>• une analyse des index cible pour les clés d'abonnement.</li> </ul> <b>simple</b> Génère les informations dans le rapport standard mais exclut les informations détaillées de la table IBMSNAP_SUBS_COLS.
<b>-tl</b> <i>n</i>	Indique l'intervalle de date (de 0 à 30 jours) des entrées à extraire de la table IBMSNAP_APPLYTRAIL. La valeur par défaut est de 3 jours.
<b>-at</b> <i>n</i>	Indique l'intervalle de date (de 0 à 30 jours) des entrées à extraire de la table IBMSNAP_APPLYTRACE de trace d'Apply. La valeur par défaut est de 3 jours.
<b>-ct</b> <i>n</i>	Indique l'intervalle de date (de 0 à 30 jours) des entrées à extraire de la table IBMSNAP_CAPTRACE. La valeur par défaut est de 3 jours.
<b>-cm</b> <i>n</i>	Indique l'intervalle de date (de 0 à 30 jours) des entrées à extraire de la table IBMSNAP_CAPMON. La valeur par défaut est de 3 jours.
<b>-sg</b> <i>n</i>	Indique l'intervalle de date (de 0 à 30 jours) des entrées à extraire de la table IBMSNAP_SIGNAL. La valeur par défaut est de 3 jours.

Tableau 25. Définitions des paramètres d'appel d'asnanalyze pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Définition
<b>-aq</b> <i>qualificatif_apply</i>	Indique le qualificatif Apply identifiant les ensembles d'abonnements spécifiques à analyser.  Vous pouvez indiquer plusieurs qualificatifs Apply. Dans ce cas, séparez les valeurs par des espaces. Si aucun qualificatif Apply n'est indiqué, tous les ensembles d'abonnements pour les alias de base de données indiqués sont analysés.
<b>-cs</b> <i>schéma_capture</i>	Indique le nom du schéma de Capture à analyser.  Avec ce paramètre, vous ne pouvez préciser qu'un seul schéma de Capture.
<b>-od</b> <i>répertoire_sortie</i>	Indique le répertoire dans lequel vous voulez stocker le rapport de l'analyseur. Il s'agit par défaut du répertoire de travail.
<b>-fn</b> <i>nomfichier_sortie</i>	Indique le nom du fichier qui contiendra la sortie du rapport de l'analyseur.  Suivez les conventions de dénomination du système d'exploitation sur lequel vous exécutez l'analyseur. Si le nom de fichier existe déjà, le fichier est écrasé. Le nom de fichier par défaut est asnanalyze.htm.
<b>-pw</b> <i>cheminfichier_mot de passe</i>	Indique le nom et le chemin d'accès du fichier des mots de passe. Si vous ne spécifiez pas ce paramètre, l'Analyseur vérifie le répertoire en cours pour le fichier asnpwd.aut.

## Exemples pour asnanalyze

Les exemples suivants illustrent comment utiliser la commande **asnanalyze**.

### Exemple 1

Pour analyser les tables de contrôle de réplication dans la base de données proddb1 :

```
asnanalyze -db proddb1
```

### Exemple 2

Pour obtenir un niveau détaillé d'analyse sur les tables de contrôle de réplication dans les bases de données proddb1 et proddb2 :

```
asnanalyze -db proddb1 proddb2 -la detailed
```

### Exemple 3

Pour analyser les deux derniers jours d'informations depuis les tables IBMSNAP\_APPLYTRAIL, IBMSNAP\_APPLYTRACE, IBMSNAP\_CAPTRACE, IBMSNAP\_CAPMON et IBMSNAP\_SIGNAL dans les bases de données proddb1 et proddb2 :

```
asnanalyze -db proddb1 proddb2 -t1 2 -at 2 -ct 2 -cm 2 -sg 2
```

### Exemple 4

Pour obtenir un niveau simple d'analyse sur les deux derniers jours d'informations depuis les tables IBMSNAP\_APPLYTRAIL, IBMSNAP\_APPLYTRACE, IBMSNAP\_CAPTRACE, IBMSNAP\_CAPMON et IBMSNAP\_SIGNAL dans les bases de données proddb1 et proddb2 pour les qualificatifs Apply qual1 et qual2 seulement :

```
asnanalyze -db proddb1 proddb2 -la simple -t1 2 -at 2 -ct 2 -cm 2 -sg 2
-aq qual1 qual2 -od c:\mydir -fn anzout -pw c:\SQLLIB
```

Cet exemple de commande écrit la sortie de l'analyseur dans un fichier anzout dans le répertoire c:\mydir et utilise les informations de mot de passe figurant dans le répertoire c:\SQLLIB.

### Exemple 5

Pour analyser un schéma de Capture déterminé :

```
asnanalyze -db proddb1 proddb2 -cs BSN
```

### Exemple 6

Pour afficher l'aide sur la commande :

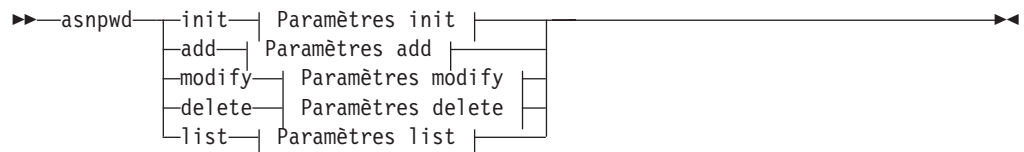
```
asnanalyze
```

## asnpwd : création et gestion des fichiers de mots de passe

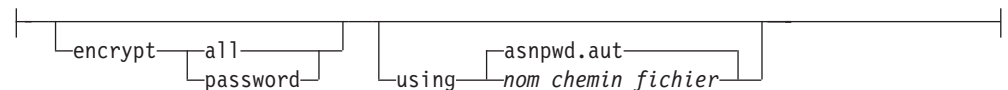
Utilisez la commande **asnpwd** pour créer et modifier les fichiers sous Linux, UNIX et Windows. Exécutez cette commande sur la ligne de commande ou dans un script de shell.

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.

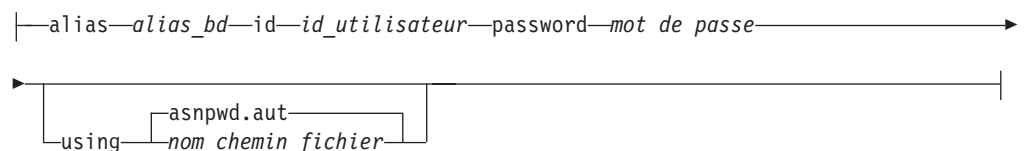
### Syntaxe



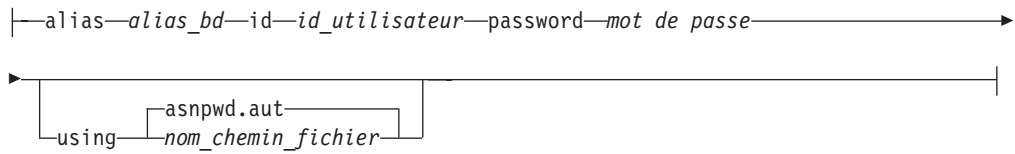
### Paramètres init :



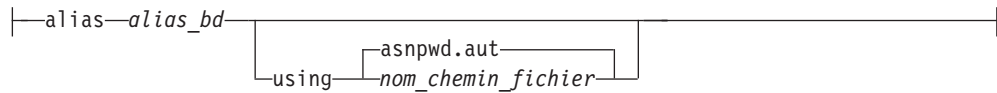
### Paramètres add :



### Paramètres modify :



### Paramètres delete :



### Paramètres list :



## Paramètres

Le tableau 26 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 26. Définitions des paramètres d'appel de la commande **asnpwd** pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Définition
<b>init</b>	Indiquez ce paramètre pour créer un fichier de mot de passe vide. Cette commande échouera si vous indiquez le paramètre <b>init</b> avec un fichier de mot de passe déjà existant.

Tableau 26. Définitions des paramètres d'appel de la commande `asnpwd` pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Définition
<b>add</b>	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 <b>add</b> avec une entrée déjà existante dans le fichier de mot de passe. Utilisez le paramètre <b>modify</b> pour modifier une entrée existante dans le fichier de mot de passe.
<b>modify</b>	Indiquez ce paramètre pour modifier le mot de passe ou l'ID utilisateur d'une entrée dans le fichier de mot de passe.
<b>delete</b>	Indiquez ce paramètre pour supprimer une entrée du fichier de mot de passe.
<b>list</b>	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 <b>encrypt password</b> . Les mots de passe ne sont jamais affichés par la commande <b>list</b> .
<b>encrypt</b>	Indique les entrées d'un fichier qui doivent être chiffrées.  <b>all (valeur par défaut)</b> 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.  <b>password</b> 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.
<b>using</b> <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:\sql11\b\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 <b>init</b> 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.
<b>alias</b> <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.
<b>id</b> <i>id_utilisateur</i>	Indique l'ID utilisateur qui a accès à la base de données.
<b>password</b> <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 :

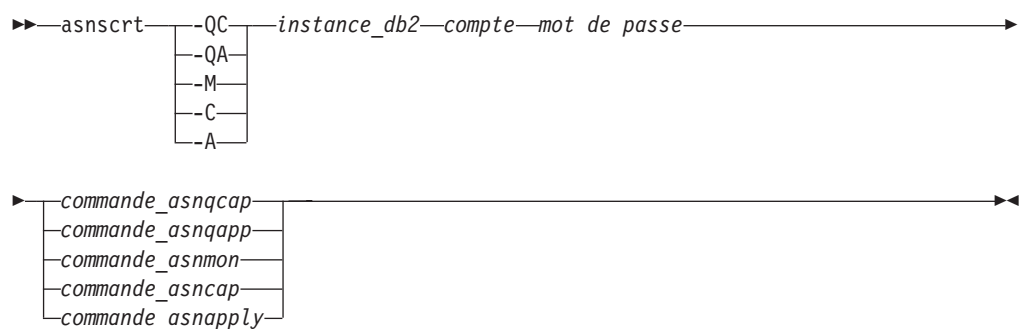
```
asnpwd LIST USING pass1.aut
Alias : SAMPLE ID : chglocalid
Nombre d'entrées : 1
```

---

## 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**, **asnsncap** et **asnsnapply**. Exécutez la commande **asnsCRT** sur le système d'exploitation Windows.

### Syntaxe



## Paramètres

Le tableau 27 définit les paramètres d'appel de la commande **asnsrct**.

Tableau 27. définition des paramètres d'appel de la commande **asnsrct** pour les systèmes d'exploitation Windows

Paramètre	Définition
<b>-QC</b>	Indique que vous lancez un programme Q Capture.
<b>-QA</b>	Indique que vous lancez un programme Q Apply.
<b>-M</b>	Indique que vous lancez un programme Moniteur d'alertes de réplication.
<b>-C</b>	Indique que vous lancez un programme Capture.
<b>-A</b>	Indique que vous lancez un programme Apply.
<b>db2_instance</b>	Indique l'instance DB2 utilisée pour identifier un service de réplication DB2 unique. L'instance DB2 peut contenir jusqu'à huit caractères.
<b>account</b>	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, nom_domaine\nom_compte).
<b>password</b>	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.
<b>asnqcap_command</b>	<p>Indique la commande <b>asnqcap</b> complète pour lancer un programme Q capture. Utilisez la syntaxe documentée de la commande <b>asnqcap</b> avec les paramètres <b>asnqcap</b>.</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 <b>capture_path</b> via la commande <b>asnqcap</b>. Si la variable DB2PATH est définie et que vous indiquez un paramètre <b>capture_path</b>, le paramètre <b>capture_path</b> remplace la variable DB2PATH.</p> <p>La commande <b>asnsrct</b> ne valide pas la syntaxe des paramètres <b>asnqcap</b> saisis.</p>
<b>asnqapp_command</b>	<p>Indique la commande <b>asnqapp</b> complète pour lancer un programme Q apply. Utilisez la syntaxe documentée de la commande <b>asnqapp</b> avec les paramètres <b>asnqapp</b> 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 <b>apply_path</b> via la commande <b>asnqapp</b>. Si la variable DB2PATH est définie et que vous indiquez un paramètre <b>apply_path</b>, le paramètre <b>apply_path</b> remplace la variable DB2PATH. La commande <b>asnsrct</b> ne valide pas la syntaxe des paramètres <b>asnqapp</b> saisis.</p>

Tableau 27. définition des paramètres d'appel de la commande `asnscri` pour les systèmes d'exploitation Windows (suite)

Paramètre	Définition
<code>asnmon_command</code>	<p>Indique la commande <code>asnmon</code> complète pour lancer un programme Moniteur d'alertes de réplication. Utilisez la syntaxe documentée de la commande <code>asnmon</code> avec les paramètres <code>asnmon</code>.</p> <p>Si la variable d'environnement <code>DB2PATH</code> n'est pas définie, vous devez indiquer un emplacement pour les fichiers journaux en incluant le paramètre <code>monitor_path</code> via la commande <code>asnmon</code>. Si la variable <code>DB2PATH</code> est définie et que vous indiquez un paramètre <code>chemin_monitor</code>, ce dernier remplace la variable <code>DB2PATH</code>.</p> <p>La commande <code>asnscri</code> ne valide pas la syntaxe des paramètres <code>asnmon</code> saisis.</p>
<code>asnscap_command</code>	<p>Indique la commande <code>asnscap</code> complète pour lancer un programme Capture. Utilisez la syntaxe documentée de la commande <code>asnscap</code> avec les paramètres <code>asnscap</code> appropriés.</p> <p>Si la variable d'environnement <code>DB2PATH</code> n'est pas définie, vous devez indiquer un emplacement pour les fichiers de travail en incluant le paramètre <code>capture_path</code> via la commande <code>asnscap</code>. Si la variable <code>DB2PATH</code> est définie et que vous indiquez un paramètre <code>capture_path</code>, le paramètre <code>capture_path</code> remplace la variable <code>DB2PATH</code>.</p> <p>La commande <code>asnscri</code> ne valide pas la syntaxe des paramètres <code>asnscap</code> saisis.</p>
<code>asnapply_command</code>	<p>Indique la commande <code>asnapply</code> complète pour lancer un programme Apply. Utilisez la syntaxe documentée de la commande <code>asnapply</code> avec les paramètres <code>asnapply</code> appropriés.</p> <p>Si la variable d'environnement <code>DB2PATH</code> n'est pas définie, vous devez indiquer l'emplacement pour les fichiers journaux en incluant le paramètre <code>apply_path</code> via la commande <code>asnapply</code>. Si la variable <code>DB2PATH</code> est définie et que vous indiquez un paramètre <code>apply_path</code>, le paramètre <code>apply_path</code> remplace la variable <code>DB2PATH</code>.</p> <p>La commande <code>asnscri</code> ne valide pas la syntaxe des paramètres <code>asnapply</code> 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:\sqllib\bin\asncap 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 ◄
```

ALL

### Paramètres

Le tableau 28, à la page 289 définit les paramètres d'appel de la commande **asnsdrop**.

Tableau 28. définitions des paramètres d'appel de la commande `asnsdrop` pour les systèmes d'exploitation Windows

Paramètre	Définition
<code>nom_service</code>	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.
<code>ALL</code>	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 29 définit le paramètre d'appel de la commande **asnslist**.

*Tableau 29. définition du paramètre d'appel de la commande asnslist pour les systèmes d'exploitation Windows*

Paramètre	Définition
<b>details</b>	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)

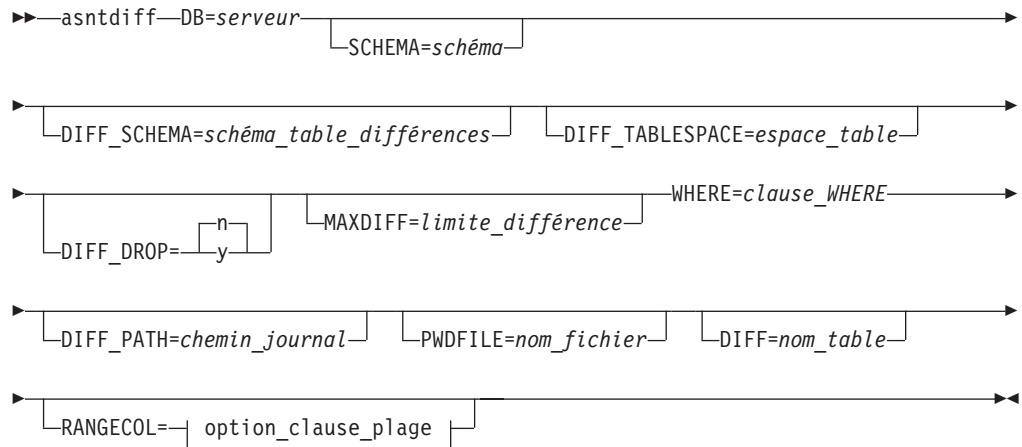
La commande **asntdiff** permet de comparer deux tables relationnelles et de générer une liste des différences entre les deux. Exécutez la commande **asntdiff** à l'invite du système d'exploitation ou dans un script de shell.

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 294. Pour obtenir des informations sur l'option de commande **asntdiff -f** qui

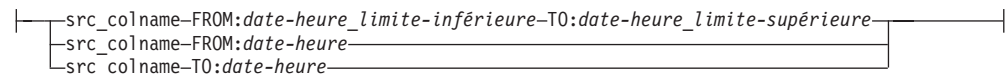
permet de comparer des tables ne faisant pas partie de la réplication à l'aide d'un fichier en entrée, voir «Option de commande `asntdiff -f` (fichier en entrée)», à la page 301.

Les tables que vous comparez peuvent se trouver sous DB2 for Linux, UNIX, Windows, DB2 for z/OS ou DB2 for System i.

## Syntaxe



### option\_clause\_plage :



## Paramètres

Le tableau 30 définit les paramètres d'appel de la commande `asntdiff`.

Tableau 30. Définitions des paramètres d'appel de la commande `asntdiff` pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Définition
<code>DB=serveur</code>	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 :  <b>réplication Q</b> Nom du serveur Q Capture qui contient la table <code>IBMQREP_SUBS</code> .  <b>réplication SQL</b> Nom du serveur de contrôle Apply qui contient la table <code>IBMSNAP_SUBS_MEMBR</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>	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 <code>asntdiff</code> a été exécutée.

Tableau 30. Définitions des paramètres d'appel de la commande *asntdiff* pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Définition
<b>DIFF_DROP</b> =y/n	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 <b>asntdiff</b> en crée une.  <b>n (par défaut)</b> La table des différences sera utilisée en l'état et les lignes existantes seront supprimées.  <b>y</b> La table des différences sera supprimée et recréée.
<b>MAXDIFF</b> =limite_différence	Indique le nombre maximal de différences que la commande <b>asntdiff</b> doit traiter avant de s'arrêter. La valeur par défaut est 10000.
<b>WHERE</b> =clause_WHERE	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 :  <b>réplication Q</b> 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.  <b>réplication SQL</b> 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.
<b>DIFF_PATH</b> =chemin_journal	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 ("").
<b>PWDFILE</b> =nom_fichier	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 <b>asnpwd</b> ). La commande <b>asntdiff</b> 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.
<b>DIFF</b> =nom_table	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 <b>DIFF_SCHEMA</b> , la table des différences s'appelle <i>ASN.ASNDDIFF</i> .



Tableau 30. Définitions des paramètres d'appel de la commande `asntdiff` pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Définition
Clause <b>RANGECOL</b>	<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> <b>FROM:</b> <i>date-heure</i> <b>limite-inférieure</b> <b>TO:</b> <i>date-heure</i> <b>limite-supérieure</b></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><b>A faire :</b> Les mots clés <b>FROM:</b> et <b>TO:</b> sont obligatoires et doivent tous deux être suivis de deux points (:).</p> <p><i>nomcol_src</i> <b>FROM:</b> <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> <b>TO:</b> <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><b>Recommandation :</b> 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><b>Restriction :</b> 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: '2008-03-10-0.00.00.00000'
TO: '2007-04-12-00.00.00.00000'"
```

### 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: '2008-03-10-12.00.00.00000'
TO: '2008-03-11-12.00.00.00000'"
```

---

## asntdiff : comparaison des données dans les tables source et cible (z/OS)

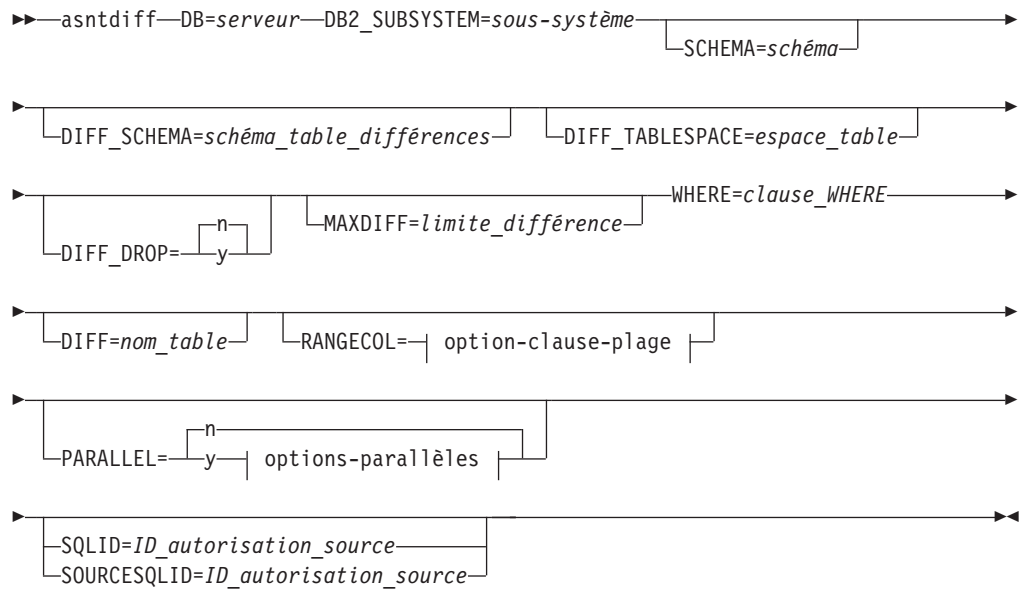
La commande **asntdiff** permet de comparer deux tables relationnelles et de générer une liste des différences entre les deux. Exécutez la commande **asntdiff** avec JCL ou dans une invite de commande des services système UNIX (USS) ou dans un script de shell.

La présente rubrique décrit l'utilisation sous z/OS :

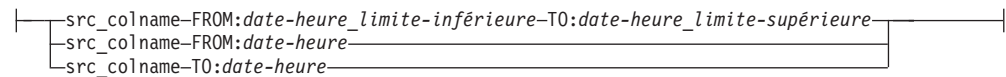
- Pour plus de détails sur l'exécution de **asntdiff** sous Linux, UNIX et Windows, voir «asntdiff : comparaison des données dans les tables source et cible (Linux, UNIX, Windows)», à la page 290.
- Pour plus d'informations sur l'option de commande **asntdiff -f**, qui vous permet de comparer des tables qu'elles soient impliquées ou non dans la réplication, voir «Option de commande **asntdiff -f** (fichier en entrée)», à la page 301.
- 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 229.

Les tables que vous comparez peuvent résider sur DB2 for z/OS, DB2 pour Linux, UNIX, Windows ou DB2 pour System i.

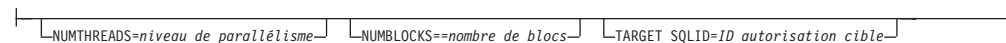
## Syntaxe



### option-clause-plage :



### options-parallèles :



## Paramètres

Le tableau 31, à la page 296 définit les paramètres d'appel de la commande `asntdiff`.

Tableau 31. 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><b>réplication Q</b> Nom de l'emplacement du serveur Q Capture qui contient la table <code>IBMQREP_SUBS</code>.</p> <p><b>réplication SQL</b> Nom de l'emplacement du serveur de contrôle Apply qui contient la table <code>IBMSNAP_SUBS_MEMBR</code>.</p> <p><b>Restriction :</b> 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><b>n (par défaut)</b> La table des différences sera utilisée en l'état et les lignes existantes seront supprimées.</p> <p><b>y</b> 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 31. Définitions des paramètres d'appel de la commande *asntdiff* pour les systèmes d'exploitation z/OS (suite)

Paramètre	Définition
<b>WHERE</b> = <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><b>réplication Q</b> 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><b>réplication SQL</b> 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><b>Restriction :</b> 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>
<b>DIFF</b> = <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 <b>DIFF_SCHEMA</b>, la table des différences s'appelle ASN.ASNTDIFF.</p>

Tableau 31. Définitions des paramètres d'appel de la commande *asntdiff* pour les systèmes d'exploitation z/OS (suite)

Paramètre	Définition
Clause <b>RANGECOL</b>	<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 :  AAAA-MM-JJ-HH.MM.SS.mmmmm. 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> <b>FROM:</b> <i>date-heure</i> <b>limite-inférieure</b> <b>TO:</b>  <i>date-heure</i> <b>limite-supérieure</b></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><b>A faire :</b> Les mots clés <b>FROM:</b> et <b>TO:</b> sont obligatoires et doivent tous deux être suivis de deux points (:).</p> <p><i>nomcol_src</i> <b>FROM:</b> <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> <b>TO:</b> <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><b>Recommandation :</b> Pour des performances optimales, assurez-vous qu'un index de la colonne source est spécifié dans la clause 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.</p> <p><b>Restriction :</b> Le paramètre RANGECOL n'est pas valide pour l'option <b>asntdiff -f</b> (fichier en entrée). Vous pouvez utiliser une clause WHERE SQL dans le fichier en entrée pour obtenir des résultats similaires.</p>

Tableau 31. Définitions des paramètres d'appel de la commande `asntdiff` pour les systèmes d'exploitation z/OS (suite)

Paramètre	Définition
<code>PARALLEL=y/n</code>	<p>Spécifie si l'utilitaire <code>asntdiff</code> 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><b>n (par défaut)</b> L'utilitaire <code>asntdiff</code> utilise le mode en série.</p> <p><b>y</b> L'utilitaire <code>asntdiff</code> utilise le mode parallèle. Pour plus de détails sur les exigences d'installation, les autorisations obligatoires et les restrictions, voir «Exécution de l'utilitaire <code>asntdiff</code> en mode parallèle (sous z/OS)», à la page 229.</p>
<code>NUMTHREADS= nombre_d'unités_d'exécution</code>	<p>Spécifie le nombre d'unités d'exécution que l'utilitaire <code>asntdiff</code> 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 <code>MAXTHREADS</code> spécifiée dans <code>BPXPRMXX</code> soit supérieure au nombre spécifié d'unités d'exécution. En outre, configurez <code>DB2 ZPARMS CTHREAD</code>, <code>IDFORE</code> et <code>IDBACK</code> pour autoriser chacune des unités d'exécution créées à se connecter à <code>DB2</code>.</p>
<code>NUMBLOCKS=nombre_de_blocs</code>	<p>Spécifie le nombre de partitions selon lequel l'utilitaire <code>asntdiff</code> divise les tables source et cible (c'est-à-dire, les ensembles de résultats des paramètres <code>SOURCE_SELECT</code> et <code>TARGET_SELECT</code>) pour comparaison parallèle. Une valeur de 0 (par défaut) signifie que l'utilitaire détermine automatiquement le nombre de blocs.</p>
<code>SQLID=ID_autorisation</code>	<p>Utilisez ce paramètre lorsque <code>asntdiff</code> 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 <code>asntdiff</code> n'est pas autorisé à créer des tables. La valeur du paramètre <code>SQLID</code> est utilisée en tant que schéma pour la table des différences si vous ne spécifiez pas explicitement un schéma à l'aide du paramètre <code>DIFF_SCHEMA</code>.</p>
<code>SOURCE_SQLID=ID_autorisation</code>	<p>Lorsque vous utilisez <code>asntdiff</code> 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 <code>asntdiff</code> ne dispose pas de l'autorisation nécessaire.</p>
<code>TARGET_SQLID=ID_autorisation</code>	<p>Lorsque vous utilisez <code>asntdiff</code> 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 <code>asntdiff</code> 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: '2008-03-10-0.00.00.000000'"
TO: '2007-04-12-00.00.00.000000'"
```

#### 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: '2008-03-10-12.00.00.000000'"
TO: '2008-03-11-12.00.00.000000'"
```

#### 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 32 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 294 ou «`asntdiff` : comparaison des données dans les tables source et cible (Linux, UNIX, Windows)», à la page 290.

Tableau 32. 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 <b>asntdiff -f</b> .
<b>SOURCE_SERVER=</b> <i>nom_serveur_source</i>	Spécifie l'alias de la base de données où se trouve la table source.
<b>TARGET_SERVER=</b> <i>nom_serveur_cible</i>	Spécifie l'alias de la base de données où se trouve la table cible.
<b>SOURCE_SELECT=</b> <i>instruction_sélection_source</i> <b>TARGET_SELECT=</b> <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 <b>asntdiff</b> 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"
```

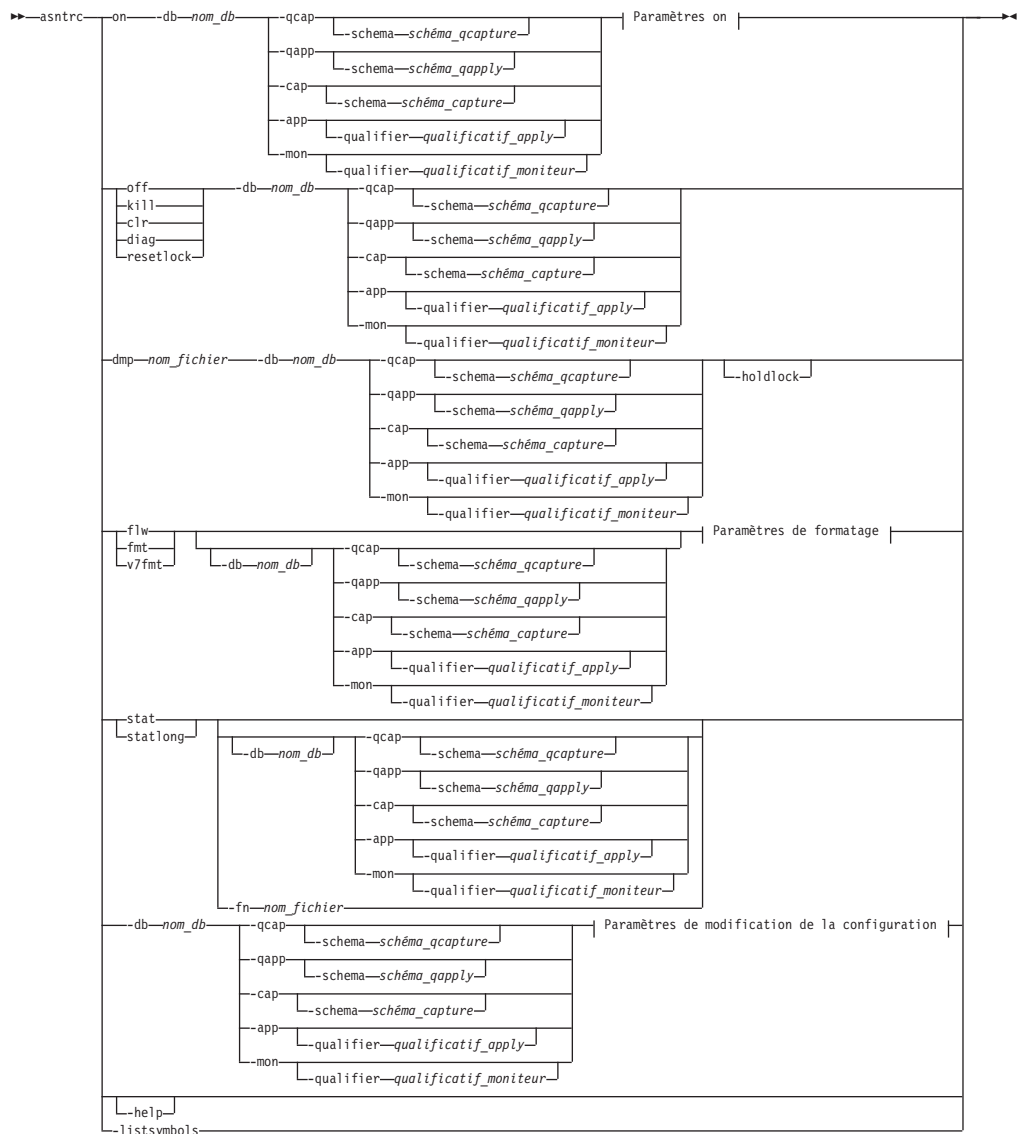
---

## asntrc : utilisation de la fonction trace de réplication

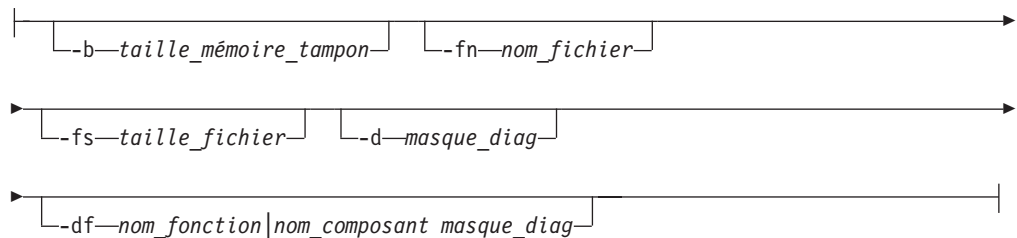
Utilisez la commande **asntrc** pour exécuter l'utilitaire trace sous Linux, UNIX, Windows et UNIX System Services (USS) sous z/OS. La fonction trace consigne les informations sur le flux de programme depuis les programmes Q Capture, Q Apply, Capture, Apply, et Replication Alert Monitor. Vous pouvez 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.

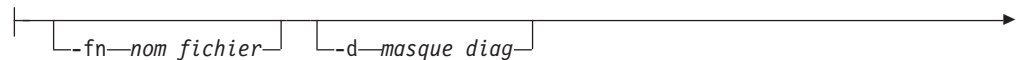
### Syntaxe



### 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 33 définit les paramètres d'appel de la commande **asntrc**.

Tableau 33. définition des paramètres d'appel de la commande **asntrc** pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS

Paramètre	Définition
<b>on</b>	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.
<b>-db</b> <i>nom_db</i>	Indique le nom de la base de données à tracer : <ul style="list-style-type: none"> <li>• Indique le nom du serveur Q Capture pour le programme Q Capture à tracer.</li> <li>• Indique le nom du serveur Q Apply pour le programme Q Apply à tracer.</li> <li>• Indique le nom du serveur de contrôle Capture pour le programme Capture à tracer.</li> <li>• Indique le nom du serveur de contrôle Apply pour le programme Apply à tracer.</li> <li>• Indique le nom du serveur de contrôle Moniteur pour le programme Moniteur d'alertes de réplication à tracer.</li> </ul>
<b>-qcap</b>	Indique qu'un programme Q Capture doit être tracé. Le programme Q Capture est identifié par le paramètre <b>-schema</b> .
<b>-schema</b> <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 <b>-qcap</b> .
<b>-qapp</b>	Indique qu'un programme Q Apply doit être tracé. Le programme Q Apply est identifié par le paramètre <b>-schema</b> .
<b>-schema</b> <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 <b>-qapp</b> .
<b>-cap</b>	Indique qu'un programme Capture doit être tracé. Le programme Capture est identifié par le paramètre <b>-schema</b> .
<b>-schema</b> <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 <b>-cap</b> .

Tableau 33. définition des paramètres d'appel de la commande *asnlrc* pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>-app</b>	Indique qu'un programme Apply doit être tracé. Le programme Apply est identifié par le paramètre <b>-qualifier</b> .
<b>-qualifier</b> <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 <b>-app</b> .
<b>-mon</b>	Indique qu'un programme Moniteur d'alertes de réplication doit être tracé. Le programme Moniteur d'alertes de réplication est identifié par le paramètre <b>-qualifier</b> .
<b>-qualifier</b> <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 <b>-mon</b> .
<b>off</b>	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.
<b>kill</b>	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 <b>off</b> .
<b>clr</b>	Indiquez ce paramètre pour effacer une mémoire tampon. Ce paramètre efface le contenu de la mémoire tampon, mais laisse cette dernière activée.
<b>diag</b>	Indiquez ce paramètre pour visualiser les paramètres du filtre pendant l'exécution de l'utilitaire trace.
<b>resetlock</b>	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.
<b>dmp</b> <i>nom_fichier</i>	Indiquez ce paramètre pour écrire le contenu actuel de la mémoire tampon de trace dans un fichier.
<b>-holdlock</b>	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.
<b>flw</b>	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 33. définition des paramètres d'appel de la commande *asnrtrc* pour les systèmes d'exploitation Linux, UNIX, Windows et z/OS (suite)

Paramètre	Définition
<b>fmt</b>	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.
<b>v7fmt</b>	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.
<b>stat</b>	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.
<b>statlong</b>	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.
<b>-fn</b> <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.
<b>-help</b>	Affiche les paramètres de la commande valide avec des descriptions.
<b>-listsymbols</b>	Affiche la fonction et les identificateurs de composants valides à utiliser avec le paramètre <b>-df</b> .
<b>-b</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.
<b>-fs</b> <i>taille_fichier</i>	Indique la taille limite (en octets) du fichier d'informations de trace en miroir.

Tableau 33. définition des paramètres d'appel de la commande `asnlrc` 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><b>Important :</b> 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 `asnlrc`

Les exemples suivants illustrent comment utiliser la commande `asnlrc`. 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 :

```
asnlrc 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 **asncap** :

```
asntrc -db DSN6 -schema JAY -cap
asncap 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 **asnapply** :

```
asntrc -db SVL_LAB_DSN6 -qualifier MYQUAL -app
asnapply 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 34 définit les paramètres d'appel de la commande **asntrep**.

Tableau 34. Définitions des paramètres d'appel de la commande **asntrep** pour les systèmes d'exploitation Linux, UNIX et Windows

Paramètre	Définition
<b>DB=serveur</b>	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 : <b>réplification Q</b> La valeur est le nom du serveur Q Capture qui contient la table IBMQREP_SUBS. <b>réplification SQL</b> La valeur est le nom du serveur de contrôle Apply qui contient la table IBMSNAP_SUBS_MEMBR. <b>z/OS</b> La valeur de ce paramètre est un nom d'emplacement.
<b>DB2_SUBSYSTEM=sous-système</b>	<b>z/OS</b> Indique le nom du sous-système dans lequel vous exécutez l'utilitaire asntrep.
<b>SCHEMA=schéma</b>	Indique le schéma des tables de contrôle Q Capture pour la réplification Q ou les tables de contrôle Apply pour la réplification SQL.
<b>DIFF_SCHEMA=schéma_table_différences</b>	Indique le schéma qui qualifie la table des différences. Il s'agit par défaut de ASN.

Tableau 34. Définitions des paramètres d'appel de la commande *asntrep* pour les systèmes d'exploitation Linux, UNIX et Windows (suite)

Paramètre	Définition
<b>DIFF_TABLESPACE</b> = <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 <b>asntrep</b> a été exécutée.
<b>WHERE</b> = <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><b>réplication Q</b> 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><b>réplication SQL</b> 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>
<b>DIFF_PATH</b> = <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 ("").
<b>PWDFILE</b> = <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 <b>asnpwd</b> ). L'utilitaire <i>asntrep</i> recherche le mot de passe dans le répertoire qui est spécifié par le paramètre <b>DIFF_PATH</b> . Si aucune valeur n'est spécifiée pour le paramètre <b>DIFF_PATH</b> , la commande recherche le fichier de mot de passe dans le répertoire où la commande a été exécutée.
<b>DIFF</b> = <i>nom_table</i>	Indique le nom de la table qui a été créée dans la base de données source en utilisant la commande <b>asntdiff</b> 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
```





---

## Chapitre 22. Commandes systèmes pour la réplication SQL (System i)

Certaines commandes de réplication sont spécifiques au système d'exploitation System i sur des serveurs System i. Vous pouvez saisir ces commandes à l'invite d'un système d'exploitation ou par un programme de ligne de commande.

Les rubriques suivantes décrivent ces commandes.

---

### ADDDPRREG : ajout d'un enregistrement DPR (System i)

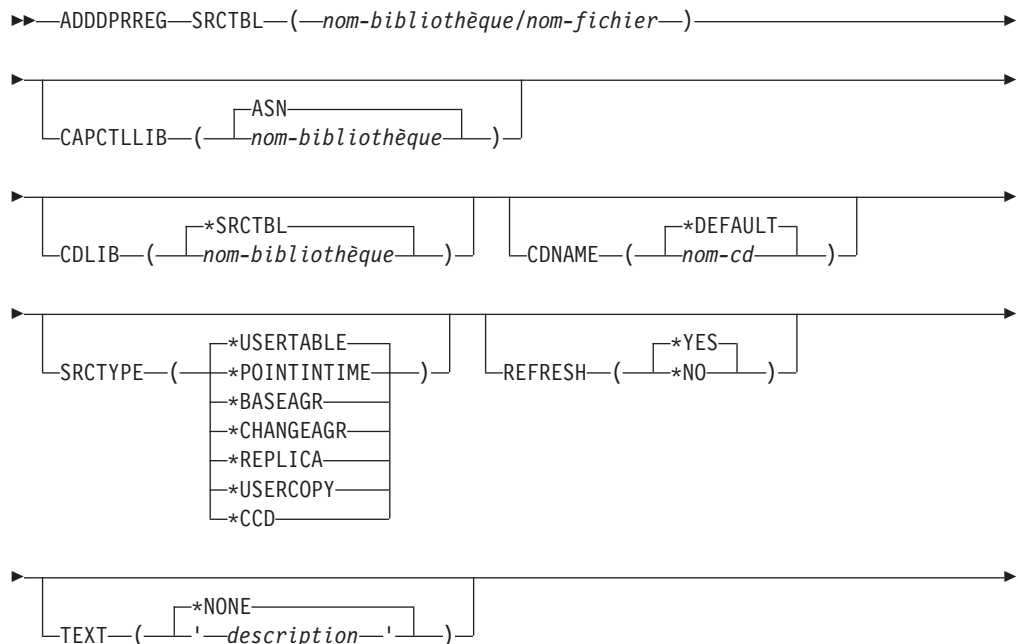
Utilisez la commande Add DPR registration (**ADDDPRREG**) pour enregistrer une table en tant que table source pour DB2 DataPropagator pour iSeries.

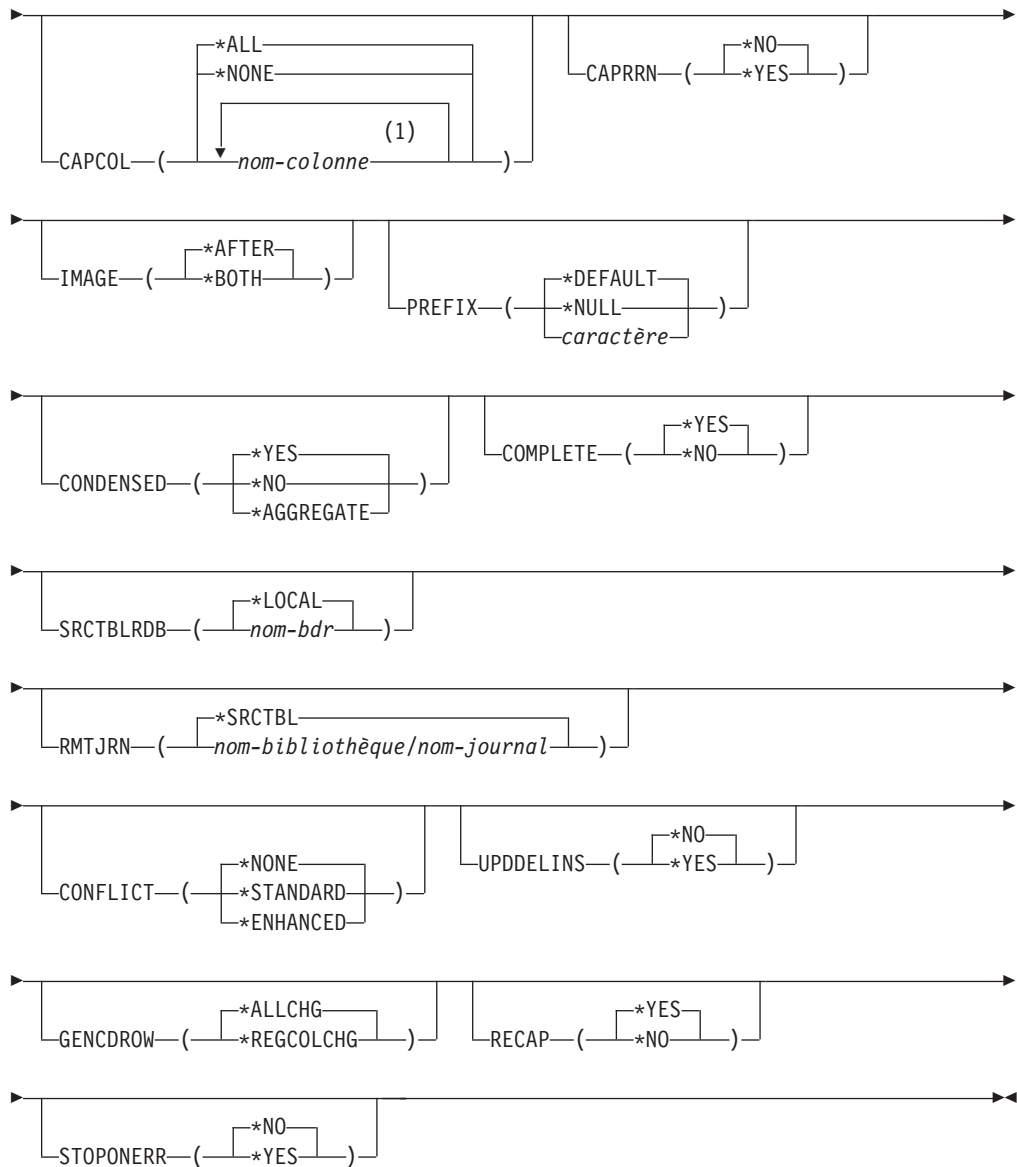
**Restriction** : Vous pouvez enregistrer une table uniquement si la bibliothèque ASN (schéma Capture) se trouve dans le même pool de mémoire secondaire (qu'il s'agisse d'un ASP de base ou indépendant) où est située la bibliothèque ASN.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

#### Syntaxe





### Remarques :

1 Vous pouvez indiquer jusqu'à 300 noms de colonnes.

Le tableau 35 dresse la liste des paramètres d'appel.

Tableau 35. Définitions des paramètres de commande ADDDPRREG pour System i

Paramètre	Définition et invites
<b>SRCTBL</b>	Indique la table que vous voulez enregistrer comme table source. Le programme Capture prend en charge tout fichier physique dans une bibliothèque ou collection System i définie en externe et dans un format unique. Ce paramètre est obligatoire.  <i>nom-bibliothèque/nom-fichier</i> Représente le nom qualifié de la table que vous souhaitez enregistrer.

Tableau 35. Définitions des paramètres de commande ADDDPREG pour System i (suite)

Paramètre	Définition et invites
<b>CAPCTLLIB</b>	<p>Spécifie le schéma Capture, qui est le nom de la bibliothèque dans laquelle résident les tables de contrôle Capture.</p> <p><b>ASN(valeur par défaut)</b> Les tables de contrôle Capture se trouvent dans la bibliothèque ASN.</p> <p><i>nom-bibliothèque</i> Nom de la bibliothèque contenant les tables de contrôle de Capture. Vous pouvez créer cette bibliothèque à l'aide de la commande <b>CRTDPRTBL</b>, avec le paramètre <b>CAPCTLLIB</b>.</p>
<b>CDLIB</b>	<p>Indique la bibliothèque dans laquelle est créée la table des données de modification (CD) pour cette source enregistrée.</p> <p><b>*SRCTBL(valeur par défaut)</b> Crée la table CD dans la bibliothèque dans laquelle réside la table source.</p> <p><i>nom-bibliothèque</i> Crée la table CD dans le nom de la bibliothèque spécifié.</p>
<b>CDNAME</b>	<p>Indique le nom de la table des données de modification (CD).</p> <p><b>ASN (valeur par défaut)</b> Crée la table CD avec le nom par défaut, basé sur l'horodatage en cours. Par exemple, si l'horodatage en cours est le 23 janvier 2002 à 09:58:26, le nom par défaut est ASN020123095826CD.</p> <p><i>nom-cd</i> Crée la table CD avec ce nom spécifié.</p>

Tableau 35. Définitions des paramètres de commande ADDDPREG pour System i (suite)

Paramètre	Définition et invites
SRCTYPE	<p>Indique le type de table source que vous enregistrez. Sélectionnez un type de source basé sur votre configuration de réplication :</p> <ul style="list-style-type: none"> <li>• Utilisez la valeur par défaut USERTABLE pour une distribution des données de base ou une configuration de la consolidation des données.</li> <li>• Utilisez REPLICA pour une configuration update-anywhere.</li> <li>• Utilisez POINTINTIME, BASEAGR, CHANGEAGR, USERCOPY, ou CCD si vous avez une configuration multi-niveaux et que vous voulez que la table cible soit une source pour un niveau suivant dans votre configuration de réplication.</li> </ul> <p>Si vous enregistrez une table cible existante en tant que source, l'enregistrement échoue si la table cible ne contient pas les colonnes de table IBMSNAP indiquées par le type de source spécifié.</p> <p><b>*USERTABLE (valeur par défaut)</b>            Une table de base de données d'utilisateur qui constitue le type le plus commun de table enregistrée. La table ne peut pas contenir de colonnes commençant par un identifiant de colonne DB2 DataPropagator pour System i de IBMSNAP ou IBMQSQ.</p> <p><b>*POINTINTIME</b>            Une copie d'une table des points de cohérence est une table cible avec un contenu qui correspond à tout ou partie du contenu de la table source et inclut la colonne système DB2 DataPropagator pour System i (IBMSNAP_LOGMARKER), qui identifie quand une ligne particulière a été insérée ou mise à jour sur le serveur de contrôle source. La table doit contenir la colonne d'horodatage IBMSNAP_LOGMARKER et peut en option contenir une colonne INTEGER appelée IBMQSQ_RRN.</p> <p><b>*BASEAGR</b>            Une copie cumulée de base, qui contient les données cumulées à certains intervalles depuis une table utilisateur ou une table des points de cohérence. Cette table cumulée de base doit contenir les colonnes d'horodatage IBMSNAP_HLOGMARKER et IBMSNAP_LLOGMARKER.</p> <p><b>*CHANGEAGR</b>            Une table de copie cumulée des modification, qui contient les cumuls de données basés sur les modifications enregistrées pour une table source. La table doit contenir les colonnes d'horodatage IBMSNAP_HLOGMARKER et IBMSNAP_LLOGMARKER.</p> <p><b>*REPLICA</b>            Une table cible pour un abonnement de réplication. Enregistrez ce type de table afin que les modification de la table cible soient répliquées de nouveau dans la table source originale. Cette table ne peut pas contenir de colonnes de système DB2 DataPropagator pour System i ni de colonnes commençant par l'identifiant de colonne IBMSNAP ou IBMQSQ DB2 DataPropagator pour System i. La table contient toutes les colonnes de la table source originale.</p> <p><b>*USERCOPY</b>            Une table cible dont le contenu correspond à tout ou partie du contenu d'une table source. La table de copie utilisateur contient uniquement des colonnes de données utilisateur.</p>

Tableau 35. Définitions des paramètres de commande ADDDPREG pour System i (suite)

Paramètre	Définition et invites
SRCTYPE (suite)	<p><b>*CCD</b></p> <p>Une table de modification cohérente des données (CCD), qui contient des données cohérentes de transaction à partir de la table source. La table doit contenir des colonnes définies comme suit :</p> <ul style="list-style-type: none"> <li>• IBMSNAP_INTENTSEQ CHAR(10) FOR BIT DATA NOT NULL</li> <li>• IBMSNAP_OPERATION CHAR(1) NOT NULL</li> <li>• IBMSNAP_COMMITSEQ CHAR(10) FOR BIT DATA NOT NULL</li> <li>• IBMSNAP_LOGMARKER TIMESTAMP NOT NULL</li> </ul>
REFRESH	<p>Indique si la fonction de régénération intégrale est activée. Vous pouvez utiliser cette valeur pour désactiver la capacité du programme Apply afin d'effectuer une régénération intégrale de la base de données source.</p> <p><b>*YES(valeur par défaut)</b> Les régénérations intégrales sont autorisées.</p> <p><b>*NO</b> Les régénérations intégrales ne sont pas autorisées.</p> <p>Si la table cible est une table d'agrégation de base ou une table d'agrégation des modifications, vous devez définir ce paramètre sur *No.</p>
TEXT	<p>Indique la description textuelle associée à cet enregistrement.</p> <p><b>*NONE(valeur par défaut)</b> Aucune description n'est associée à l'entrée.</p> <p><i>description</i> La description textuelle de cet enregistrement. Vous pouvez saisir un maximum de 50 caractères et devez mettre le texte entre guillemets simples.</p>
CAPCOL	<p>Indique la colonne pour laquelle les modifications sont saisies pour cette table enregistrée.</p> <p><b>*ALL(valeur par défaut)</b> Les modifications sont capturées pour toutes les colonnes.</p> <p><b>*NONE</b> Les modifications ne sont pas capturées pour cette table. Utilisez cette valeur pour indiquer que vous voulez cette table enregistrée uniquement pour une régénération intégrale. La table des données de modification (CD) n'est pas créée avec cette table enregistrée, et le programme Capture ne capturera pas les modifications pour la table.</p> <p><i>nom-colonne</i> Les noms de colonnes pour lesquels les modifications sont capturées. Vous pouvez indiquer jusqu'à 300 noms de colonnes. Séparez les noms de colonne par des espaces.</p>

Tableau 35. Définitions des paramètres de commande ADDDPRREG pour System i (suite)

Paramètre	Définition et invites
<b>CAPRRN</b>	<p>Indique si le numéro d'enregistrement relatif (RRN) de chaque enregistrement modifié est capturé.</p> <p><b>*NO(valeur par défaut)</b> Le numéro d'enregistrement relatif n'est pas capturé.</p> <p><b>*YES</b> Le numéro d'enregistrement relatif est capturé. Une colonne supplémentaire appelée IBMQSQ_RRN est créée dans la table des données de modification (CD).  Ne définissez ce paramètre sur *YES que s'il n'y a pas de clés uniques dans la table source.</p>
<b>IMAGE</b>	<p>Indique si la table des données de modification (CD) contient des images avant et des images après des modifications de la table source. Cela s'applique globalement à toutes les colonnes spécifiées dans le paramètre des colonnes Capture (CAPCOL).</p> <p>Ce paramètre <b>IMAGE</b> n'est pas valide lorsque le paramètre CAPCOL est défini sur *NONE.</p> <p>La table source doit être journalisée avec les images *BOTH même si vous indiquez *AFTER dans ce paramètre.</p> <p><b>*AFTER(valeur par défaut)</b> Le programme Capture enregistre uniquement les images après de la table source dans la table CD.</p> <p><b>*BOTH</b> Le programme Capture enregistre les images après et avant de la table source dans la table CD.</p>
<b>PREFIX</b>	<p>Indique le caractère du préfixe identifiant les noms de colonne des images avant dans la table des données de modification (CD). Vous devez veiller à ce qu'aucun des caractères des noms de colonne enregistrés de la table source ne commence avec ce caractère de préfixe.</p> <p><b>*DEFAULT(valeur par défaut)</b> Le préfixe par défaut (@) est utilisé.</p> <p><b>*NULL</b> Aucune image avant n'est capturée. Cette valeur n'est pas valide si le paramètre <b>IMAGE</b> est défini sur *BOTH.</p> <p><i>character</i> Un caractère alphabétique unique qui est valide dans un nom d'objet.</p>
<b>CONDENSED</b>	<p>Indique si la table source est condensée. Une table condensée contient des données actuelles avec une ligne maximum pour chaque valeur de clé primaire dans la table.</p> <p><b>*YES(valeur par défaut)</b> La table source est condensée.</p> <p><b>*NO</b> La table source n'est pas condensée.</p> <p><b>*AGGREGATE</b> Le type de table source est *BASEAGR (agrégat de base) ou *CHANGEAGR (agrégat de modification). Si cette valeur est utilisée, vous devez définir le paramètre COMPLETE sur *No</p>

Tableau 35. Définitions des paramètres de commande ADDDPREG pour System i (suite)

Paramètre	Définition et invites
<b>COMPLETE</b>	<p>Indique si la table source est complète, ce qui signifie que la table contient une ligne pour chaque valeur de clé primaire intéressante.</p> <p><b>*YES (valeur par défaut)</b> La table source est complète.</p> <p><b>*NO</b> La table source n'est pas complète.</p>
<b>SRCTBLRDB</b>	<p>Indique si vous voulez utiliser la journalisation distante dans laquelle la table source et le journal distant résident dans des systèmes différents. Utilisez ce paramètre pour spécifier l'emplacement de la table source.</p> <p><b>*LOCAL (valeur par défaut)</b> La table source réside localement (sur l'ordinateur sur lequel vous exécutez la commande ADDDPREG).</p> <p><i>nom-bdr</i> Le nom de la base de données relationnelle dans laquelle réside la table source. Vous pouvez utiliser la commande Work with RDB Directory Entries (<b>WRKRDBDIRE</b>) pour trouver ce nom de base de données relationnelles.</p>
<b>RMTJRN</b>	<p>Indique le nom du journal distant lorsque le nom de ce journal et le nom du journal sur le système source sont différents. Vous devez émettre cette commande à partir du système dans lequel réside le journal distant.</p> <p><b>*SRCTBL (valeur par défaut)</b> Le nom du journal distant est le même que le nom du journal de la table source.</p> <p><i>nom-bibliothèque/nom-journal</i> Le nom qualifié de la bibliothèque et du journal qui se trouvent sur ce système et qui est utilisé pour la journalisation de la table source distante.</p> <p>Vous ne pouvez indiquer un nom de journal distant que si vous avez spécifié l'emplacement distant de la table source à l'aide du paramètre <b>SRCTBLRDB</b>.</p>
<b>CONFLICT</b>	<p>Indique le niveau de conflit qui est utilisé par le programme Apply lorsqu'il détecte des conflits dans un abonnement de réplication.</p> <p><b>*NONE (valeur par défaut)</b> Pas de détection de conflit.</p> <p><b>*STANDARD</b> Détection de conflit modérée. Le programme Apply recherche des conflits dans les lignes qui sont déjà capturées dans la tables des données de modification (CD) de réplication.</p> <p><b>*ENHANCED</b> Détection de conflit améliorée. Cette option assure la meilleure intégrité des données parmi toutes les réplication et les tables source.</p>

Tableau 35. Définitions des paramètres de commande ADDDPREG pour System i (suite)

Paramètre	Définition et invites
UPDELINS	<p>Détermine comment le programme Capture stocke les données source mises à jour dans la table des données de modification (CD).</p> <p><b>*NO (valeur par défaut)</b> Le programme Capture stocke chaque modification de la source dans une seule ligne de la table CD.</p> <p><b>*YES</b> Le programme Capture stocke chaque modification de la source en utilisant deux lignes de la table CD, une pour la suppression, et une pour l'insertion. Le programme Apply traite d'abord la ligne de suppression et ensuite la ligne d'insertion.</p>
GENCDROW	<p>Indique si le programme Capture capture les modifications de toutes les lignes dans la table source.</p> <p><b>*ALLCHG(valeur par défaut)</b> Le programme Capture capture les modifications de toutes les lignes dans la table source (y compris les modifications dans les colonnes non enregistrées) et ajoute ces modifications dans la table des données de modification (CD).</p> <p><b>*REGCOLCHG</b> Le programme Capture capture les modification uniquement si les modifications ont lieu dans les colonnes enregistrées. Le programme Capture ajoute ensuite ces lignes à la table de modification des données.</p> <p>Vous ne pouvez pas indiquer *REGCOLCHG si le paramètre CAPCOL est défini sur *ALL ou *NONE.</p>
RECAP	<p>Indique si les modifications apportées par le programme Apply sont recapturées par le programme Capture.</p> <p><b>*YES(valeur par défaut)</b> Les modifications apportées sur la table source par le programme Apply sont capturées et saisies dans la table des données de modification (CD).</p> <p><b>*NO</b> Les modifications apportées sur la table source par le programme Apply ne sont pas capturées et par conséquent, n'apparaissent pas dans la table des données de modification (CD). Vous devez utiliser cette option lors de l'enregistrement des tables de type REPLICA.</p>
STOPONERR	<p>Indique si le programme Capture s'arrête lorsqu'il rencontre une erreur.<sup>1</sup></p> <p><b>*NO(valeur par défaut)</b> Le programme Capture ne s'arrête pas quand il rencontre une erreur. Le programme Capture émet des messages, désactive l'enregistrement qui a provoqué l'erreur, puis continue le traitement.</p> <p><b>*YES</b> Le programme Capture émet des messages et s'arrête lorsqu'il rencontre une erreur.</p>



Tableau 35. Définitions des paramètres de commande ADDDPRREG pour System i (suite)

Paramètre	Définition et invites
<b>Remarque :</b>	
1.	<p>Si ce paramètre est défini sur Yes (Y), le travail de journalisation de Capture s'arrête pendant les autres travaux de journalisation continuent de s'exécuter. Si ce paramètre est défini sur No (N), le travail de journalisation de Capture arrête le fichier d'enregistrement qui contient l'erreur.</p> <p>Ce paramètre définit également les colonnes dans les lignes de la table des registres. La colonne STATE est définie sur 'S' et la colonne STATE_INFO est définie sur 200Axxxx où xxxx est le code anomalie. Pour redéfinir l'enregistrement sur l'état Action ('A'), procédez comme suit :</p> <ul style="list-style-type: none"> <li>• Corrigez le message ASN200A. Consultez la documentation System i appropriée pour l'action corrigée.</li> <li>• Utilisez le centre de réplication ou la commande System i STRSQL pour définir les colonnes dans la ligne de la table IBMSNAP_REGISTER. Définissez la colonne STATE sur 'A', et la colonne STATE_INFO sur null.</li> <li>• Si le programme Capture est en cours d'exécution, émettez la commande INZDPRCAP pour réinitialiser la réplication de données pour ce journal.</li> </ul>

## Exemples pour ADDDPRREG

Les exemples suivants illustrent la manière d'utiliser la commande **ADDDPRREG**.

### Exemple 1 :

Par exemple, pour enregistrer une table source nommée EMPLOYEE depuis la bibliothèque HR dans le schéma Capture par défaut :

```
ADDDPRREG SRCTBL(HR/EMPLOYEE)
```

### Exemple 2 :

Pour enregistrer la table source nommée EMPLOYEE depuis la bibliothèque HR dans le schéma Capture BSN et créer une table CD nommée CDEMPLOYEE dans la bibliothèque HRCDLIB :

```
ADDDPRREG SRCTBL(HR/EMPLOYEE) CAPCTLLIB(BSN) CDLIB(HRCDLIB) CDNAME(CDEMPLOYEE)
```

### Exemple 3 :

Pour enregistrer une table source avec un type de point de cohérence nommé SALES depuis la bibliothèque DEPT sous le schéma Capture BSN :

```
ADDDPRREG SRCTBL(DEPT/SALES) CAPCTLLIB(BSN) SRCTYPE(*POINTINTIME)
```

### Exemple 4 :

Pour enregistrer une table source nommée SALES depuis la bibliothèque DEPT et indiquer que la table CD contient que les images avant et images après des modifications de la table source :

```
ADDDPRREG SRCTBL(DEPT/SALES) IMAGE(*BOTH)
```

### Exemple 5 :

Pour enregistrer une table source nommée SALES depuis la bibliothèque DEPT de la base de données relationnelle nommée RMTRDB1 en utilisant les journaux distants :

ADDDPRREG SRCTBL(DEPT/SALES) SRCTBLRDB(RMTRDB1) RMTJRN(RMTJRNLIB/RMTJRN)

**Exemple 6 :**

Pour enregistrer la table source EMPLOYEE depuis la bibliothèque HR et capturer uniquement les modifications pour les colonnes EMPNO, NAME, DEPT, et NETPAY :

ADDDPRREG SRCTBL(HR/EMPLOYEE) CAPCOL(EMPNO NAME DEPT NETPAY)

## ADDDPRSUB : ajout d'un ensemble d'abonnements DPR (System i)

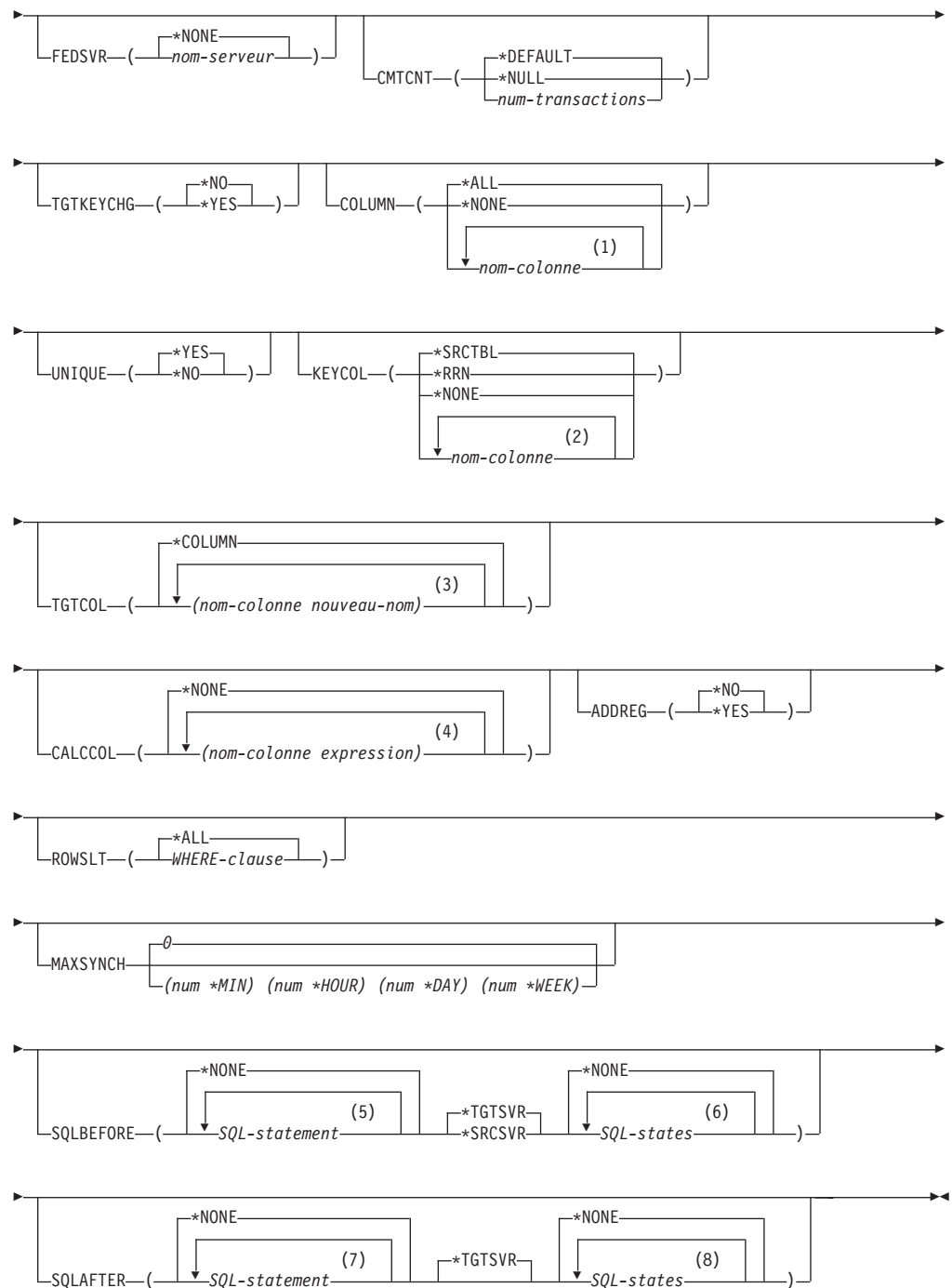
Utilisez la commande Add DPR subscription set (**ADDDPRSUB**) pour créer un ensemble d'abonnements défini avec un membre ou sans membre.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe





**Remarques :**

- 1 Vous pouvez indiquer jusqu'à 300 noms de colonnes.
- 2 Vous pouvez indiquer jusqu'à 120 noms de colonnes.
- 3 Vous pouvez indiquer jusqu'à 300 noms de colonnes.
- 4 Vous pouvez indiquer jusqu'à 100 noms et expressions de colonnes.
- 5 Vous pouvez indiquer jusqu'à 3 instructions SQL.
- 6 Vous pouvez spécifier jusqu'à 10 SQLSTATES.

- 7 Vous pouvez indiquer jusqu'à 3 instructions SQL.
- 8 Vous pouvez spécifier jusqu'à 10 SQLSTATES.

Le tableau 36 dresse la liste des paramètres d'appel.

Tableau 36. Définitions des paramètres de commande **ADDDPRSUB** pour System i

Paramètre	Définition et invites
<b>APYQUAL</b>	<p>Indique le qualificatif Apply identifiant quel programme Apply traite cet ensemble d'abonnements. Les ensembles d'abonnements sous un qualificatif Apply s'exécutent dans un travail distinct. Ce paramètre est obligatoire.</p> <p><i>apply-qualifier</i> Nom du qualificatif Apply.</p>
<b>SETNAME</b>	<p>Indique le nom de l'ensemble d'abonnements. Ce paramètre est obligatoire.</p> <p><i>nom-ensemble</i> Nom de l'ensemble d'abonnements. Le nom de l'ensemble d'abonnements que vous saisissez doit être unique pour le qualificatif Apply indiqué, sinon la commande <b>ADDDPRSUB</b> génère une erreur. Dans la mesure où le programme Apply gère l'ensemble des tables cible en tant que groupe, lorsqu'une table cible échoue, pour quelque raison que ce soit, tout l'ensemble d'abonnements échoue également.</p>
<b>SRCTBL</b>	<p>Spécifie le nom de la table source utilisée pour copier les informations dans votre ensemble d'abonnements. Vous devez enregistrer cette table sur le serveur de contrôle Capture avant que cette table ne devienne un membre d'un ensemble d'abonnements. Ce paramètre est obligatoire.</p> <p><b>*NONE (valeur par défaut)</b> Cet ensemble d'abonnements n'a pas de membre source. A utiliser lors de la création d'un ensemble d'abonnements sans membre.</p> <p><i>nom-bibliothèque/nom-fichier</i> Le nom qualifié de la table source. A utiliser lors de la création d'un ensemble d'abonnements avec un membre.</p>
<b>TGTTBL</b>	<p>Spécifie le nom de la table cible. La table cible est automatiquement créée si vous réglez le paramètre <b>CRTTGTTBL</b> sur *YES et que la table cible n'existe pas déjà. Ce paramètre est obligatoire.</p> <p><b>*NONE (valeur par défaut)</b> Cet ensemble d'abonnements n'a pas de membre cible. A utiliser lors de la création d'un ensemble d'abonnements sans membre.</p> <p><i>nom-bibliothèque/nom-fichier</i> Le nom qualifié de la table cible. A utiliser lors de la création d'un ensemble d'abonnements avec un membre.</p>
<b>CTLSVR</b>	<p>Spécifie le nom de la base de données relationnelle du système contenant les tables de contrôle Apply.</p> <p><b>*LOCAL (par défaut)</b> Les tables de contrôle Apply résident localement (sur l'ordinateur sur lequel vous exécutez la commande <b>ADDDPRSUB</b>).</p> <p><i>nom-bdr</i> Le nom de la base de données relationnelle où résident les tables de contrôle Apply. Vous pouvez utiliser la commande Work with RDB Directory Entries (<b>WRKRBDIRE</b>) pour trouver ce nom.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
SRCSV	<p>Spécifie le nom de la base de données relationnelle du système contenant les tables de contrôle Capture.</p> <p><b>*LOCAL (par défaut)</b>            Les table source est enregistrée sur la machine locale (sur l'ordinateur sur lequel vous exécutez la commande <b>ADDDPRSUB</b>).</p> <p><i>nom-bdr</i>            Le nom de la base de données relationnelle où résident les tables de contrôle Capture. Vous pouvez utiliser la commande Work with RDB Directory Entries (<b>WRKRDBDIRE</b>) pour trouver ce nom.</p>
TGTYPE	<p>Spécifie le type de table cible. Après avoir créé une table cible en tant qu'un de ces types, vous pouvez utiliser cette valeur de paramètre sur le paramètre <b>SRCTBL</b> de la commande Add DPR Registration (<b>ADDDPRREG</b>) pour enregistrer cette table cible comme table source pour une réplication multi-niveaux.</p> <p><b>*USERCOPY (valeur par défaut)</b>            Cette table cible est une copie utilisateur qui est une table cible avec un contenu qui correspond à tout ou partie du contenu d'une table source. Une copie utilisateur est gérée comme une copie des points de cohérence mais ne contient pas les colonnes système DB2 DataPropagator pour System i que l'on trouve dans la table cible des points de cohérence.</p> <p>Cette valeur n'est pas valide lorsqu'une valeur *RRN est spécifiée dans le paramètre <b>KEYCOL</b>.</p> <p>La table que vous avez spécifiée avec le paramètre <b>SRCTBL</b> doit avoir l'un des types suivants : base de données utilisateur, copie des points de cohérence, ou table de modification cohérente des données (CCD).</p> <p><b>Important</b> : Si la table cible existe déjà; DB2 DataPropagator pour System i ne consigne pas automatiquement les modifications qui y sont apportées. Vous devez commencer la journalisation en dehors de DB2 DataPropagator pour System i.</p> <p><b>*POINTINTIME</b>            La table cible est une copie des points de cohérence. Une copie des points de cohérence est une table cible avec un contenu qui correspond à tout ou partie du contenu de la table source et inclut la colonne système DB2 DataPropagator pour System i (<b>IBMSNAP_LOGMARKER</b>), qui identifie quand une ligne particulière a été insérée ou mise à jour sur le serveur de contrôle Capture.</p> <p><b>*BASEAGR</b>            La table cible est une copie agrégée de base qui est une table cible qui contient des données agrégés (calculées) à partir d'une table source. La table source pour une cible agrégée doit être une table utilisateur ou une table des points de cohérence. Cette table cible doit contenir les colonnes d'horodatage du système <b>IBMSNAP_HLOGMARKER</b> et <b>IBMSNAP_LLOGMARKER</b>.</p> <p><b>*CHANGEAGR</b>            La table cible est une copie agrégée de change qui est une table cible qui contient des données agrégées (calculées) à partir d'une table de données de modification. Cette table cible est créée avec les colonnes d'horodatage système <b>IBMSNAP_HLOGMARKER</b> et <b>IBMSNAP_LLOGMARKER</b>.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
<b>TGTYPE</b> (suite)	<p><b>*CCD</b> La table est une table de modification cohérente des données (CCD), qui est une table cible créée pour un regroupement de données dans la table des données de modification (CD) et la table des unités d'oeuvre (UOW). Une table CCD fournit des données cohérentes de transaction pour le programme Apply et comprend les colonnes suivantes :</p> <ul style="list-style-type: none"> <li>• IBMSNAP_INTENTSEQ</li> <li>• IBMSNAP_OPERATION</li> <li>• IBMSNAP_COMMITSEQ</li> <li>• IBMSNAP_LOGMARKER</li> </ul> <p><b>*REPLICA</b> La table cible est une table de réplication, utilisée uniquement pour la réplication update-anywhere. La réplication reçoit les modifications de la table source maître, et les modifications dans la table cible de réplication sont propagées en retour sur la table source maître. Une table de réplication est automatiquement enregistrée en tant que table source.</p>
<b>TIMING</b>	<p>Spécifie le type de timing (planification) que le programme Apply utilise pour traiter l'ensemble d'abonnements.</p> <p><b>*INTERVAL (valeur par défaut)</b> Le programme Apply traite l'ensemble d'abonnements à un intervalle de temps spécifique (par exemple, une fois par jour).</p> <p><b>*EVENT</b> Le programme Apply traite l'ensemble d'abonnements lorsqu'un événement spécifique se produit.</p> <p><b>*BOTH</b> Le programme Apply traite l'ensemble d'abonnements à un intervalle spécifique ou lorsqu'un événement se produit, selon celui qui a lieu en premier.</p>
<b>EVENT</b>	<p>Spécifie un événement. L'événement que vous saisissez doit correspondre à un nom d'événement dans la table IBMSNAP_SUBS_EVENT).</p> <p><b>*NONE (valeur par défaut)</b> Aucun événement n'est utilisé.</p> <p><i>nom-événement</i> Une chaîne de caractères unique qui représente un événement décrit dans la table IBMSNAP_SUBS_EVENT.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
<b>INTERVAL</b>	<p>Spécifie l'intervalle de temps (semaine, jours, heures et minutes) depuis l'heure de début jusqu'à l'heure de début entre les régénérations de la copie cible. Il s'agit d'une valeur en deux parties. La première partie est un chiffre, et la deuxième partie l'unité de temps :</p> <p><b>*MIN</b> Minutes</p> <p><b>*HOUR</b> Heures</p> <p><b>*DAY</b> Jours</p> <p><b>*WEEK</b> Semaines</p> <p>Vous pouvez spécifier des combinaisons de chiffres avec des unités de temps. par exemple ((2 *WEEK) (3 *DAY) (35 *MIN)) indique un intervalle de temps de deux semaines, trois jours et 35 minutes. Si vous spécifiez plusieurs occurrences de la même unité de temps, c'est la dernière occurrence qui est utilisée.</p>
<b>ACTIVATE</b>	<p>Indique si l'ensemble d'abonnements est actif Le programme Apply ne traite pas cet ensemble d'abonnements sauf si le paramètre est défini sur *YES.</p> <p><b>*YES (valeur par défaut)</b> L'ensemble d'abonnements est actif.</p> <p><b>*NO</b> L'ensemble d'abonnements n'est pas actif.</p>
<b>CRTTGTTBL</b>	<p>Indique si la table cible (ou la vue) est créée.</p> <p><b>*YES (valeur par défaut)</b> Crée la table cible (ou la vue) si elle n'existe pas. Sinon, la table cible ou la vue devient la cible, et le format de cette table ou vue existante est vérifié si la valeur du paramètre <b>CHKFMT</b> est sur *YES. Un index supplémentaire, avec les valeur que vous avez indiquées par les paramètres <b>UNIQUE</b> et <b>KEYCOL</b>, est créé pour une table cible, si aucun index de ce type n'existe actuellement. La commande échoue si une table cible existante contient des lignes qui enfreignent les conditions de l'index supplémentaire.</p> <p><b>*NO</b> Ne crée pas la table cible ou la vue. Vous devez créer la table ou la vue avec les attributs adéquats avant de lancer le programme Apply.</p> <p>Si la table ou la vue existe et que vous définissez <b>CHKFMT</b> sur *YES, la commande <b>ADDDPRSUB</b> garantir que le format de la table existante correspond à la définition de l'ensemble d'abonnements que vous avez paramétrée. Si <b>CHKFMT</b> est *NO, vous devez vous assurer que le format de la table existante correspond à la définition de l'ensemble d'abonnements.</p> <p><b>Important</b> : Si la table ou la vue existe déjà, DB2 DataPropagator pour System i ne consigne pas automatiquement les modifications apportées à l'objet existant. Vous devez commencer la journalisation en dehors de DB2 DataPropagator pour System i.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
<b>CHKFMT</b>	<p>Indique si DB2 DataPropagator pour System i contrôle l'ensemble d'abonnements et la table cible pour s'assurer que les colonnes correspondent. Ce paramètre est ignoré si le paramètre <b>CRTTGTTBL</b> est *YES ; ce paramètre est également ignoré si le paramètre <b>CRTTGTTBL</b> est défini sur *NO et que la table cible n'existe pas.</p> <p><b>*YES (valeur par défaut)</b> DB2 DataPropagator pour System i vérifie que les colonnes définies pour cet ensemble d'abonnements correspondent aux colonnes de la table cible. Cette commande échoue si une non concordance est détectée.</p> <p><b>*NO</b> DB2 DataPropagator pour System i ignore les différences entre l'ensemble d'abonnements et la table cible existante. Vous devez veiller à ce que la table cible soit compatible avec l'ensemble d'abonnements.</p>
<b>CAPCTLLIB</b>	<p>Spécifie le schéma Capture, qui est le nom de la bibliothèque dans laquelle résident les tables de contrôle Capture. Ces tables de contrôle Capture traitent la source de cet ensemble d'abonnements.</p> <p><b>ASN (valeur par défaut)</b> Les tables de contrôle Capture se trouvent dans la bibliothèque ASN.</p> <p><i>nom-bibliothèque</i> Nom d'une bibliothèque contenant les tables de contrôle de Capture. Il s'agit de la bibliothèque dans laquelle a été enregistrée la table source.</p>
<b>TGTCCLIB</b>	<p>Indique la bibliothèque de contrôle cible.</p> <p><b>*CAPCTLLIB (valeur par défaut)</b> La bibliothèque de contrôle cible est la même bibliothèque que celle dans laquelle se trouvent les tables de contrôle Capture.</p> <p><i>nom-bibliothèque</i> Nom d'une bibliothèque contenant les tables de contrôle cible.</p> <p>Si vous utilisez une table cible comme source d'un autre ensemble d'abonnements (comme une table de données de modification cohérente CCD), cette valeur de paramètre est le schéma de Capture lorsque cette table est utilisée comme source.</p>
<b>FEDSVR</b>	<p>Indique si un système de base de données fédérée est la source de cet ensemble d'abonnements.</p> <p><b>*NONE (valeur par défaut)</b> Le serveur source n'est pas un système de base de données fédérée.</p> <p><i>nom-serveur</i> Le nom du système de base de données fédérée pour cet ensemble d'abonnements (pour les sources relationnelles non DB2).</p>



Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
<b>CMTCNT</b>	<p>Indique le décompte de validation, qui correspond au nombre de transactions que traite le programme Apply avant une validation.</p> <p><b>*DEFAULT (valeur par défaut)</b> La commande détermine la valeur à utiliser. Si le <b>TGTTYPE</b> est défini sur <b>*REPLICA</b>, alors le <b>CMTCNT</b> est égal à zéro (0). Si le <b>TGTTYPE</b> est différent de <b>*REPLICA</b>, le <b>CMTCNT</b> est null.</p> <p><b>*NULL</b> L'ensemble d'abonnements est en lecture seule. Le programme Apply extraira les ensembles de réponses pour les membres de l'ensemble d'abonnements, un membre à la fois, jusqu'à ce que toutes les données aient été traitées et émettra une validation unique pour tout l'ensemble d'abonnements.</p> <p><i>num-transactions</i> Indique le nombre de transactions traitées avant que le programme Apply ne valide les modifications. Ce paramètre n'est valide que si le paramètre <b>TGTTYPE</b> est défini sur <b>*REPLICA</b>.</p>
<b>TGTKEYCHG</b>	<p>Indique comment le programme Apply gère les mises à jour qui ont lieu dans les colonnes source qui font partie des colonnes clé cible de la table cible. Ce paramètre fonctionne en conjonction avec le paramètre <b>USEDELINS</b> de la commande <b>ADDDPRREG</b> :</p> <ul style="list-style-type: none"> <li>• Si <b>USEDELINS</b> est YES et <b>TGTKEYCHG</b> est YES, les mises à jour ne sont pas autorisées.</li> <li>• Si <b>USEDELINS</b> est YES et <b>TGTKEYCHG</b> est NO, les mises à jour sont supprimées et insèrent des paires.</li> <li>• Si <b>USEDELINS</b> est NO et <b>TGTKEYCHG</b> est YES, le programme Apply gère cette condition avec une logique spéciale.</li> <li>• Si <b>USEDELINS</b> est NO et <b>TGTKEYCHG</b> est NO, le programme Apply traite les modifications en tant que mises à jour normales.</li> </ul> <p><b>*NO (valeur par défaut)</b> Les mises à jour de la table source sont faites par étape par le programme Capture et traitées par le programme Apply dans la table cible.</p> <p><b>*YES</b> Le programme Apply met à jour la table cible en fonction des images avant de la colonne de clés cibles, ce qui signifie que le programme Apply modifie le prédicat avec les anciennes valeurs au lieu des nouvelles.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
<b>COLUMN</b>	<p>Indique les colonnes à inclure dans la table cible. les noms de colonnes doivent être non qualifiés. Sélectionnez les noms de colonne dans la liste de noms de colonnes que vous avez spécifiée avec le paramètre <b>CAPCOL</b> lorsque vous avez enregistré la table source.</p> <p>Si vous avez défini le paramètre <b>IMAGE</b> sur <b>*BOTH</b> lors de l'enregistrement de cette table, vous pouvez spécifier les noms de colonnes d'image avant. Les noms de colonnes d'image avant sont les noms de colonne originaux avec un préfixe. Ce préfixe est le caractère que vous avez spécifié dans le paramètre <b>PREFIX</b> de la commande <b>ADDDPRREG</b>.</p> <p><b>*ALL (valeur par défaut)</b> Toutes les colonnes que vous avez enregistrées dans la source sont incluses dans la table cible.</p> <p><b>*NONE</b> Aucune colonne de la table source n'est incluse dans la table cible. Vous pouvez utiliser <b>*NONE</b> lorsque vous voulez uniquement les colonnes calculées dans la table cible. Cette valeur est obligatoire si le paramètre <b>CALCCOL</b> contient des fonctions récapitulatives mais qu'aucun <b>GROUP BY</b> n'est effectué.</p> <p><i>nom-colonne</i> Les noms d'un maximum de 300 colonnes source que vous voulez inclure dans la table cible. Séparez les noms de colonne par des espaces.</p>
<b>UNIQUE</b>	<p>Indique si la table cible a des clés uniques comme indiqué par le paramètre <b>KEYCOL</b>.</p> <p><b>*YES (valeur par défaut)</b> La table cible prend en charge une modification nette par clé ; une seule ligne existe dans la table cible pour cette clé, quel que soit le nombre de modifications effectuées sur la clé.</p> <p>Cette valeur indique que la table contient des données en cours au lieu d'un historique des modifications des données. Une table condensée comprend plus d'une ligne pour chaque clé primaire dans la table et peut être utilisée pour fournir des informations actuelles pour une régénération.</p> <p><b>*NO</b> La table cible prend en charge plusieurs modifications par clé. Les modifications sont jointes à la table cible.</p> <p>Cette valeur indique que la table contient un historique des modifications des données au lieu des données en cours. Une table non condensée comprend plus d'une ligne pour chaque valeur clé dans la table et peut être utilisée pour fournir un historique des modifications des données. Une table non condensée ne peut pas fournir de données actuelles pour une régénération.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
<b>KEYCOL</b>	<p>Indique les colonnes qui décrivent la clé de la table cible. les noms de colonnes doivent être non qualifiés. Pour les tables cible *POINTINTIME, *REPLICA, et *USERCOPY (comme spécifié dans le paramètre <b>TGTYPE</b>), vous devez identifier une ou plusieurs colonnes comme clé cible pour la table cible. La clé cible est utilisée par le programme Apply pour identifier chaque ligne uniquement qui change pendant la réplication des captures de modification.</p> <p><b>*SRCTBL (valeur par défaut)</b></p> <p>Les colonnes clé de la table cible sont identiques à celles de la table source. La commande <b>ADDDPRREG</b> utilise la clé spécifiée dans la table source comme si la table source était indexée. Les colonnes clé suivantes sont utilisées :</p> <ul style="list-style-type: none"> <li>• Les colonnes clé que vous avez définies via DDS lors de la création de la table avec la commande Create Physical File (<b>CRTPF</b>)</li> <li>• Les clés primaire et unique que vous avez définies avec les instructions CREATE TABLE et ALTER TABLE SQL</li> <li>• Les clés uniques que vous avez définies avec les instructions CREATE INDEX SQL</li> </ul> <p>Si vous utilisez une colonne en tant que clé plus d'une fois et avec un ordre différent, la clé de la table cible est définie dans l'ordre croissant.</p> <p><b>*RRN</b></p> <p>La colonne clé de la table cible est la colonne IBMQSQ_RRN. La table cible est créée avec une colonne IBMQSQ_RRN, et cette colonne est utilisée comme clé. Lorsque le programme Apply s'exécute, si la table source est une table utilisateur et que la table cible est une table des points de cohérence ou une copie utilisateur, la colonne IBMQSQ_RRN dans la table cible est mise à jour avec le numéro d'enregistrement relatif de l'enregistrement associé dans la table source. Sinon, la colonne IBMQSQ_RRN dans la table cible est mise à jour avec la valeur de la colonne IBMQSQ_RRN dans la table source.</p> <p><b>*NONE</b></p> <p>La copie cible ne contient pas de clé cible. Vous ne pouvez pas spécifier *NONE si le type de la table cible est *POINTINTIME, *REPLICA, ou *USERCOPY.</p> <p><i>nom-colonne</i></p> <p>Les noms des colonnes cibles que vous voulez utiliser comme colonnes de clé cible. Vous pouvez indiquer jusqu'à 120 noms de colonnes. Séparez les noms de colonne par des espaces.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
<b>TGTCOL</b>	<p>Indique les nouveaux noms de toutes les colonnes que le programme Apply met à jour dans la table cible. Ces noms remplacent les noms de colonne extraits de la table source. Les noms de colonne doivent être non qualifiés. Si vous avez spécifié une valeur de *NONE pour le paramètre <b>COLUMN</b>, n'utilisez pas ce paramètre.</p> <p>Utilisez ce paramètre pour donner des noms significatifs aux colonnes de table cible. Indiquez chaque nom de colonne source ainsi que le nom de la colonne correspondante sur la table cible.</p> <p><b>*COLUMN (valeur par défaut)</b> Les colonnes cibles sont identiques aux colonnes que vous avez spécifiées dans le paramètre <b>COLUMN</b>.</p> <p><i>nom-colonne</i> Les noms de colonne de la table source que vous voulez modifier dans la cible. Vous pouvez indiquer jusqu'à 300 noms de colonnes.</p> <p><i>nouveau-nom</i> Les nouveaux noms des colonnes cibles. Vous pouvez indiquer jusqu'à 300 nouveaux noms de colonnes. Si vous n'utilisez pas ce paramètre, le nom de la colonne dans la table cible sera identique au nom de la colonne source.</p>
<b>CALCCOL</b>	<p>Indique la liste des colonnes définies par l'utilisateur ou calculées dans la table cible. Les noms de colonne doivent être non qualifiés. Mettez chaque paire de nom de colonne et d'expression entre parenthèses.</p> <p>Vous devez spécifier un nom de colonne pour chaque expression SQL. Si vous voulez définir une colonne en tant qu'expression SQL sans instruction GROUP BY, vous devez définir le paramètre <b>COLUMN</b> sur *NONE.</p> <p><b>*NONE (valeur par défaut)</b> Aucune colonne définie par l'utilisateur ou calculée n'est incluse dans la table cible.</p> <p><i>nom-colonne</i> Les noms de colonne des colonnes définies par l'utilisateur ou calculées dans la table cible. Vous pouvez indiquer jusqu'à 100 noms de colonnes.</p> <p><i>expression</i> Les expressions des colonnes définies par l'utilisateur ou calculées dans la table cible. Vous pouvez indiquer jusqu'à 100 expressions de colonnes SQL.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
<b>ADDREG</b>	<p>Indique si la table cible est automatiquement enregistrée comme table source. Utilisez ce paramètre pour enregistrer des tables de type cible CCD.</p> <p><b>*NO (valeur par défaut)</b> La table cible n'est pas enregistrée comme table source. DB2 DataPropagator pour System i ignore cette valeur de paramètre si le type de cible est *REPLICA. Les tables cible de réplication sont toujours enregistrées en tant que tables source.</p> <p><b>*YES</b> La table cible est enregistrée en tant que table source. Cette commande échoue si vous avez déjà enregistré la table cible.</p> <p>Ne définissez pas ce paramètre sur *YES si le type de table cible est *USERCOPY, *POINTINTIME, *BASEAGR, ou *CHANGEAGR.</p> <p>Si vous définissez le paramètre <b>CRTTGTTBL</b> sur *NO, vous devez créer la table cible avant de tenter de l'enregistrer en tant que source.</p>
<b>ROWSLT</b>	<p>Indique les prédicats devant être placés dans une clause SQL WHERE. Le programme Apply utilise ces prédicats pour déterminer les lignes dans la table de données de modification (CD) de la source à appliquer dans la table cible. Utilisez ce paramètre si vous voulez que seul un sous-ensemble des modifications de la source soit répliqué dans la table cible.</p> <p><b>*ALL (valeur par défaut)</b> Le programme Apply applique toutes les modifications de la table CD dans la table cible.</p> <p><i>WHERE-clause</i> La clause SQL WHERE qui spécifie quelles lignes de la table CD le programme Apply applique dans la table cible. Ne pas inclure le mot clé WHERE ; il est implicite dans ce paramètre. Cette clause WHERE doit être valide sur le serveur de données que vous utilisez pour exécuter la clause.</p> <p><b>Note :</b> La clause WHERE sur ce paramètre est non associée aux clauses WHERE spécifiées dans les paramètres SQLBEFORE ou SQLAFTER.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
<b>MAXSYNCH</b>	<p>Indique le nombre maximal de minutes de synchronisation. Ce paramètre représente la limite de seuil de temps utilisée pour réguler la quantité de données de modification traitée par les programmes Capture et Apply pendant un cycle d'abonnement. Vous pouvez indiquer une limite de seuil de temps à l'aide d'une valeur à deux parties. La première partie est un chiffre, et la deuxième partie l'unité de temps :</p> <p><b>*MIN</b> Minutes</p> <p><b>*HOUR</b> Heures</p> <p><b>*DAY</b> Jours</p> <p><b>*WEEK</b> Semaines</p> <p>Vous pouvez spécifier des combinaisons de chiffres avec des unités de temps. Par exemple ((1 *WEEK) (2 *DAY) (35 *MIN)) indique un intervalle de temps de deux semaines, trois jours et 35 minutes. Si vous spécifiez plusieurs occurrences de la même unité de temps, c'est la dernière occurrence qui est utilisée.</p> <p>La valeur par défaut est zéro (0), ce qui indique que toutes les données de modification doivent être appliquées.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
SQLBEFORE	<p>Indique les instructions SQL qui s'exécutent avant que le programme Apply ne régénère la table cible. Ce paramètre comporte trois éléments :</p> <p>Elément 1 : code SQL</p> <p><b>*NONE (valeur par défaut)</b> Aucune instruction SQL n'est spécifiée.</p> <p><i>SQL-statement</i> L'instruction SQL que vous voulez exécuter. Veillez à ce que la syntaxe de l'instruction SQL soit correcte. DB2 DataPropagator pour System i ne valide pas la syntaxe. De plus, vous devez utiliser les conventions de dénomination SQL adéquates. Les références du fichier SQL doivent être sous la forme LIBRARY.FILE et non selon la convention de dénomination du système (LIBRARY/FILE). Vous pouvez indiquer jusqu'à trois instructions SQL.</p> <p>Elément 2 : serveur sur lequel exécuter</p> <p><b>*TGTSVR (valeur par défaut)</b> L'instruction SQL s'exécute sur le serveur cible sur lequel est situé la table cible.</p> <p><b>*SRCSVR</b> L'instruction SQL s'exécute sur le serveur de contrôle Capture sur lequel est situé la table source.</p> <p>Elément 3 : Valeurs SQLSTATE autorisées</p> <p><b>*NONE (valeur par défaut)</b> Seule une valeur SQLSTATE de 00000 est considérée valide.</p> <p><i>SQL-states</i> Une liste de une à dix valeurs SQLSTATE autorisées. Séparez les valeurs SQLSTATE par des espaces. Une valeur SQLSTATE est un nombre hexadécimal à cinq chiffres allant de 00000 à FFFFF.</p> <p>L'instruction SQL est valide si elle termine avec une valeur SQLSTATE de 00000 ou avec les valeurs SQLSTATE autorisées que vous avez listées.</p>

Tableau 36. Définitions des paramètres de commande ADDDPRSUB pour System i (suite)

Paramètre	Définition et invites
<b>SQLAFTER</b>	<p>Indique les instructions SQL qui s'exécutent après que le programme Apply ait régénéré la table cible. Ce paramètre comporte trois éléments :</p> <p>Élément 1 : code SQL</p> <p><b>*NONE (valeur par défaut)</b> Aucune instruction SQL n'est spécifiée.</p> <p><i>SQL-statement</i> L'instruction SQL que vous voulez exécuter. Veillez à ce que la syntaxe de l'instruction SQL soit correcte. DB2 DataPropagator pour System i ne valide pas la syntaxe. De plus, vous devez utiliser les conventions de dénomination SQL adéquates. Les références du fichier SQL doivent être sous la forme LIBRARY.FILE et non selon la convention de dénomination du système (LIBRARY/FILE). Vous pouvez indiquer jusqu'à trois instructions SQL.</p> <p>Élément 2 : serveur sur lequel exécuter</p> <p><b>*TGTSVR (valeur par défaut)</b> L'instruction SQL s'exécute sur le serveur cible sur lequel est situé la table cible.</p> <p>Élément 3 : Valeurs SQLSTATE autorisées</p> <p><b>*NONE (valeur par défaut)</b> Seule une valeur SQLSTATE de 00000 est considérée valide.</p> <p><i>SQL-states</i> Une liste de une à dix valeurs SQLSTATE autorisées. Séparez les valeurs SQLSTATE par des espaces. Une valeur SQLSTATE est un nombre hexadécimal à cinq chiffres allant de 00000 à FFFFF.</p> <p>L'instruction SQL est valide si elle termine avec une valeur SQLSTATE de 00000 ou avec les valeurs SQLSTATE autorisées que vous avez listées.</p>

## Exemples pour ADDDPRSUB

Les exemples suivants illustrent comment utiliser la commande **ADDDPRSUB**.

### Exemple 1 :

Pour créer un ensemble d'abonnements nommé SETHR sous le qualificatif Apply AQHR :

```
ADDDPRSUB APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/EMPLOYEE)
TGTTBL(TGTLIB/TGTEMPL)
```

Cet ensemble d'abonnements, qui contient un membre, réplique les données de la table source enregistrée EMPLOYEE de la bibliothèque HR dans la table cible TGTEMPL de la bibliothèque TGTLIB.

### Exemple 2 :

Pour créer un ensemble d'abonnements nommé SETHR avec seulement deux colonnes, EMPNO (la clé) et NAME, depuis la table source enregistrée nommée EMPLOYEE et répliquer ces colonnes dans une table cible existante nommée TGTEMPL :



```

ADDDPRSUB APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/EMPLOYEE)
  TGTTBL(TGTLIB/TGTEMPL) CRTTGTTBL(*NO) COLUMN(EMPNO NAME) KEYCOL(EMPNO)

```

**Exemple 3 :**

Pour créer un ensemble d'abonnements nommé SETHR avec des données provenant de la table source enregistrée nommée EMPLOYEE et répliquer ces colonnes dans une table cible de type de réplication nommée TGTREPL :

```

ADDDPRSUB APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/EMPLOYEE)
  TGTTBL(TGTLIB/TGTREPL) TGTTYPE(*REPLICA)

```

**Exemple 4 :**

Pour créer un ensemble d'abonnements nommé NOMEM sans membre dans l'ensemble d'abonnements :

```

ADDDPRSUB APYQUAL(AQHR) SETNAME(NOMEM) SRCTBL(*NONE) TGTTBL(*NONE)

```

## ADDDPRSUBM : ajout d'un membre d'un ensemble d'abonnements DPR (System i)

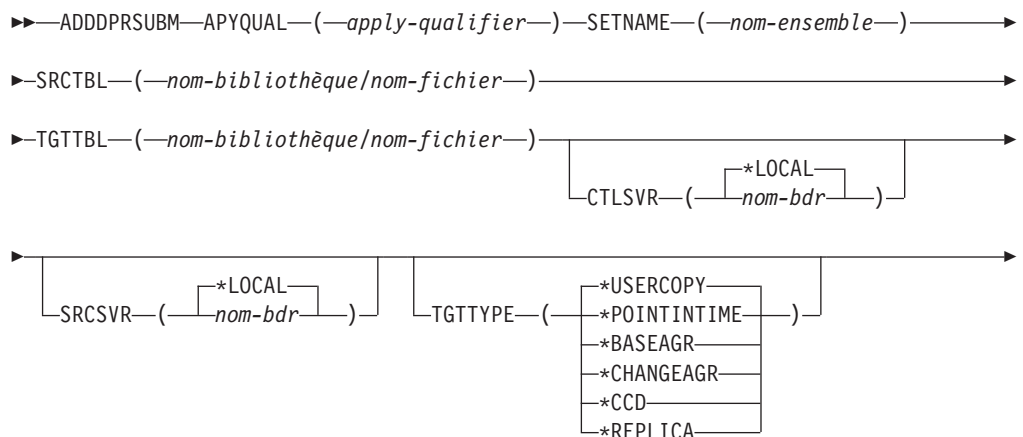
Utilisez la commande Add DPR subscription-set member (**ADDDPRSUBM**) pour ajouter un membre à un ensemble d'abonnements existant.

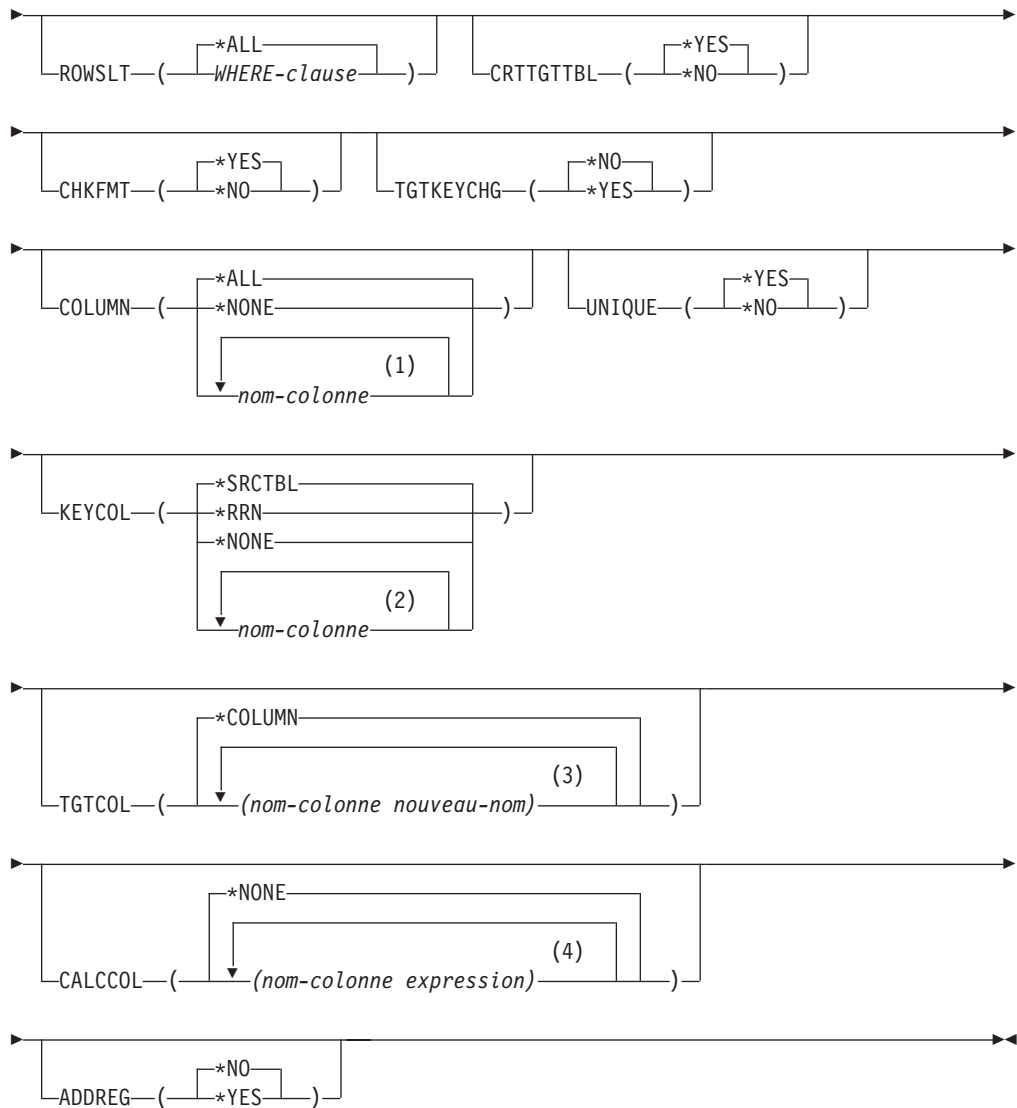
Vous pouvez créer l'ensemble d'abonnements avec la commande **ADDDPRSUB**, avec les commandes systèmes sur UNIX, Windows, ou z/OS, ou via le centre de réplication. Toutes les tables source de l'ensemble d'abonnements doivent déjà être journalisées et enregistrées avant que vous ne puissiez utiliser cette commande.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe





**Remarques :**

- 1 Vous pouvez indiquer jusqu'à 300 noms de colonnes.
- 2 Vous pouvez indiquer jusqu'à 120 noms de colonnes.
- 3 Vous pouvez indiquer jusqu'à 300 noms de colonnes.
- 4 Vous pouvez indiquer jusqu'à 100 noms et expressions de colonnes.

Le tableau 37 dresse la liste des paramètres d'appel.

Tableau 37. Définitions des paramètres de commande ADDDPRSUBM pour System i

Paramètre	Définition et invites
<b>APYQUAL</b>	Indique le qualificatif Apply identifiant quel programme Apply traite cet ensemble d'abonnements. Les ensembles d'abonnements sous un qualificatif Apply s'exécutent dans un travail distinct. Ce paramètre est obligatoire.  <i>apply-qualifier</i> Nom du qualificatif Apply.

Tableau 37. Définitions des paramètres de commande ADDDPRSUBM pour System i (suite)

Paramètre	Définition et invites
SETNAME	<p>Spécifie le nom de l'ensemble d'abonnements. Ce paramètre est obligatoire.</p> <p><i>nom-ensemble</i>            Nom de l'ensemble d'abonnements. Le nom de l'ensemble d'abonnements que vous saisissez doit être unique pour le qualificatif Apply indiqué, sinon la commande <b>ADDDPRSUBM</b> génère une erreur. Dans la mesure où le programme Apply gère l'ensemble des tables cible en tant que groupe, lorsqu'une table cible échoue, pour quelque raison que ce soit, tout l'ensemble échoue également.</p>
SRCTBL	<p>Spécifie le nom de la table qui représente la source pour ce membre de l'ensemble d'abonnements. Vous devez enregistrer cette table sur le serveur de contrôle Capture avant que cette table ne devienne un membre d'un ensemble d'abonnements. Ce paramètre est obligatoire.</p> <p><i>nom-bibliothèque/nom-fichier</i>            Le nom qualifié de la table source.</p>
TGTTBL	<p>Spécifie le nom de la table cible pour ce membre de l'ensemble d'abonnements. La table cible est automatiquement créée si vous réglez le paramètre <b>CRTTGTTBL</b> sur *YES et que la table cible n'existe pas déjà. Ce paramètre est obligatoire.</p> <p><i>nom-bibliothèque/nom-fichier</i>            Le nom qualifié de la table cible.</p>
CTLSVR	<p>Spécifie le nom de la base de données relationnelle du système contenant les tables de contrôle Apply.</p> <p><b>*LOCAL (par défaut)</b>            Les tables de contrôle Apply résident localement (sur l'ordinateur à partir duquel vous exécutez la commande <b>ADDDPRSUBM</b>).</p> <p><i>nom-bdr</i>            Le nom de la base de données relationnelle où résident les tables de contrôle Apply. Vous pouvez utiliser la commande Work with RDB Directory Entries (<b>WRKRDBDIRE</b>) pour trouver ce nom.</p>
SRCSV	<p>Spécifie le nom de la base de données relationnelle du système contenant les tables de contrôle Capture.</p> <p><b>*LOCAL (par défaut)</b>            Les table source est enregistrée sur la machine locale (sur l'ordinateur sur lequel vous exécutez la commande <b>ADDDPRSUBM</b>).</p> <p><i>nom-bdr</i>            Le nom de la base de données relationnelle où résident les tables de contrôle Capture. Vous pouvez utiliser la commande Work with RDB Directory Entries (<b>WRKRDBDIRE</b>) pour trouver ce nom.</p>

Tableau 37. Définitions des paramètres de commande ADDDPRSUBM pour System i (suite)

Paramètre	Définition et invites
TGTYPE	<p>Spécifie le type de table cible. Il s'agit des termes de réplication SQL qui décrivent le contenu de la table cible. Après avoir créé une table cible en tant qu'un de ces types, vous pouvez utiliser cette valeur de paramètre sur le paramètre <b>SRCTBL</b> de la commande Add DPR Registration (<b>ADDDPRREG</b>) pour enregistrer cette table cible comme table source.</p> <p><b>*USERCOPY (valeur par défaut)</b>            Cette table cible est une copie utilisateur qui est une table cible avec un contenu qui correspond à tout ou partie du contenu d'une table source. Une copie utilisateur est gérée comme une table de points de cohérence mais ne contient pas les colonnes système DB2 DataPropagator pour System i que l'on trouve dans la table cible des points de cohérence.</p> <p>Cette valeur n'est pas valide lorsqu'une valeur *RRN est spécifiée dans le paramètre <b>KEYCOL</b>.</p> <p>La table que vous avez spécifiée avec le paramètre <b>SRCTBL</b> doit avoir l'un des types suivants : base de données utilisateur, table des points de cohérence, ou table de modification cohérente des données (CCD).</p> <p><b>Important</b> : Si la table cible existe déjà; DB2 DataPropagator pour System i ne consigne pas automatiquement les modifications qui y sont apportées. Vous devez commencer la journalisation en dehors de DB2 DataPropagator pour System i.</p> <p><b>*POINTINTIME</b>            La table cible est une table des points de cohérence. Une table des points de cohérence est une table cible avec un contenu qui correspond à tout ou partie du contenu de la table source et inclut la colonne système DB2 DataPropagator pour System i (IBMSNAP_LOGMARKER), qui identifie quand une ligne particulière a été insérée ou mise à jour sur le serveur de contrôle Capture.</p> <p><b>*BASEAGR</b>            La table cible est une table agrégée de base qui est une table cible qui contient des données agrégées (calculées) à partir d'une table source. La table source pour une cible agrégée doit être une table utilisateur ou une table des points de cohérence. Cette table cible doit contenir les colonnes d'horodatage du système IBMSNAP_HLOGMARKER et IBMSNAP_LLOGMARKER.</p> <p><b>*CHANGEAGR</b>            La table cible est une table agrégée de modification qui est une table cible qui contient des données agrégées (calculées) à partir d'une table de données de modification. Cette table cible est créée avec les colonnes d'horodatage système IBMSNAP_HLOGMARKER et IBMSNAP_LLOGMARKER.</p>

Tableau 37. Définitions des paramètres de commande ADDDPRSUBM pour System i (suite)

Paramètre	Définition et invites
<b>TGTYPE</b> (suite)	<p><b>*CCD</b></p> <p>La table est une table de modification cohérente des données (CCD), qui est une table cible créée pour un regroupement de données dans la table des données de modification (CD) et la table des unités d'oeuvre (UOW). Une table CCD fournit des données cohérentes de transaction pour le programme Apply et comprend les colonnes suivantes :</p> <ul style="list-style-type: none"> <li>• IBMSNAP_INTENTSEQ</li> <li>• IBMSNAP_OPERATION</li> <li>• IBMSNAP_COMMITSEQ</li> <li>• IBMSNAP_LOGMARKER</li> </ul> <p><b>*REPLICA</b></p> <p>La table cible est une table de réplication, utilisée uniquement pour la réplication update-anywhere. La réplication reçoit les modifications de la table source maître, et les modifications dans la table cible de réplication sont propagées en retour sur la table source maître. Une table de réplication est automatiquement enregistrée en tant que table source.</p>
<b>ROWSLT</b>	<p>Indique les prédicats devant être placés dans une clause SQL WHERE. Le programme Apply utilise ces prédicats pour déterminer les lignes dans la table de données de modification (CD) de la source à appliquer dans la table cible. Utilisez ce paramètre si vous voulez que seul un sous-ensemble des modifications de la source soit répliqué dans la table cible.</p> <p><b>*ALL (valeur par défaut)</b></p> <p>Le programme Apply applique toutes les modifications de la table CD dans la table cible.</p> <p><i>WHERE-clause</i></p> <p>La clause SQL WHERE qui spécifie quelles lignes de la table CD le programme Apply applique dans la table cible. Ne pas inclure le mot clé WHERE ; il est implicite dans ce paramètre. Cette clause WHERE doit être valide sur le serveur de données que vous utilisez pour exécuter la clause.</p> <p><b>Note :</b> La clause WHERE sur ce paramètre est non associée aux clauses WHERE spécifiées dans les paramètres SQLBEFORE ou SQLAFTER.</p>

Tableau 37. Définitions des paramètres de commande ADDDPRSUBM pour System i (suite)

Paramètre	Définition et invites
<b>CRTTGTTBL</b>	<p>Indique si la table cible (ou la vue) est créée.</p> <p><b>*YES (valeur par défaut)</b>            Crée la table cible (ou la vue) si elle n'existe pas. Sinon, la table cible ou la vue devient la cible, et le format de cette table ou vue existante est vérifié si la valeur du paramètre <b>CHKFMT</b> est sur <b>*YES</b>. Un index supplémentaire, avec les valeur que vous avez indiquées par les paramètres <b>UNIQUE</b> et <b>KEYCOL</b>, est créé pour une table cible, si aucun index de ce type n'existe actuellement. La commande échoue si une table cible existante contient des lignes qui enfreignent les conditions de l'index supplémentaire.</p> <p><b>*NO</b>            Ne crée pas la table cible ou la vue. Vous devez créer la table ou la vue avec les attributs adéquats avant de lancer le programme Apply.</p> <p>Si la table ou la vue existe et que vous définissez <b>CHKFMT</b> sur <b>*YES</b>, la commande <b>ADDDPRSUBM</b> garantit que le format de la table existante correspond à la définition de l'ensemble d'abonnements que vous avez paramétrée. Si <b>CHKFMT</b> est <b>*NO</b>, vous devez vous assurer que le format de la table existante correspond à la définition de l'ensemble d'abonnements.</p> <p><b>Important</b> : Si la table ou la vue existe déjà, DB2 DataPropagator pour System i ne consigne pas automatiquement les modifications apportées à l'objet existant. Vous devez lancer la journalisation en dehors de DB2 DataPropagator pour System i.</p>
<b>CHKFMT</b>	<p>Indique si DB2 DataPropagator pour System i contrôle la définition du membre de l'ensemble d'abonnements et la table cible existante pour s'assurer que les colonnes correspondent. Ce paramètre est ignoré si le paramètre <b>CRTTGTTBL</b> est <b>*YES</b> ; ce paramètre est également ignoré si le paramètre <b>CRTTGTTBL</b> est défini sur <b>*NO</b> et que la table cible n'existe pas.</p> <p><b>*YES (valeur par défaut)</b>            DB2 DataPropagator pour System i vérifie que les colonnes définies pour le membre de cet ensemble d'abonnements correspondent aux colonnes de la table cible. Cette commande échoue si une non concordance est détectée.</p> <p><b>*NO</b>            DB2 DataPropagator pour System i ignore les différences entre le membre de l'ensemble d'abonnements et la table cible existante. Vous devez veiller à ce que la table cible soit compatible avec le membre de l'ensemble d'abonnements.</p>

Tableau 37. Définitions des paramètres de commande ADDDPRSUBM pour System i (suite)

Paramètre	Définition et invites
<b>TGTKEYCHG</b>	<p>Indique comment le programme Apply gère les mises à jour qui ont lieu dans les colonnes source qui font partie des colonnes clé cible de la table cible. Ce paramètre fonctionne en conjonction avec le paramètre <b>USEDELINS</b> de la commande <b>ADDDPRREG</b> :</p> <ul style="list-style-type: none"> <li>• Si <b>USEDELINS</b> est YES et <b>TGTKEYCHG</b> est YES, les mises à jour ne sont pas autorisées.</li> <li>• Si <b>USEDELINS</b> est YES et <b>TGTKEYCHG</b> est NO, les mises à jour sont supprimées et insèrent des paires.</li> <li>• Si <b>USEDELINS</b> est NO et <b>TGTKEYCHG</b> est YES, le programme Apply gère cette condition avec une logique spéciale.</li> <li>• Si <b>USEDELINS</b> est NO et <b>TGTKEYCHG</b> est NO, le programme Apply traite les modifications en tant que mises à jour normales.</li> </ul> <p><b>*NO (valeur par défaut)</b> Les mises à jour de la table source sont faites par étape par le programme Capture et traitées par le programme Apply dans la table cible.</p> <p><b>*YES</b> Le programme Apply met à jour la table cible en fonction des images avant de la colonne de clés cibles, ce qui signifie que le programme Apply modifie le prédicat avec les anciennes valeurs au lieu des nouvelles.</p>
<b>COLUMN</b>	<p>Indique les colonnes à inclure dans la table cible. les noms de colonnes doivent être non qualifiés. Sélectionnez les noms de colonne dans la liste de noms de colonnes que vous avez spécifiée avec le paramètre <b>CAPCOL</b> lorsque vous avez enregistré la table source.</p> <p>Si vous avez défini le paramètre <b>IMAGE</b> sur <b>*BOTH</b> lors de l'enregistrement de cette table, vous pouvez spécifier les noms de colonnes d'image avant. Les noms de colonnes d'image avant sont les noms de colonne originaux avec un préfixe. Ce préfixe est le caractère que vous avez spécifié dans le paramètre <b>PREFIX</b> de la commande <b>ADDDPRREG</b>.</p> <p><b>*ALL (valeur par défaut)</b> Toutes les colonnes que vous avez enregistrées dans la source sont incluses dans la table cible.</p> <p><b>*NONE</b> Aucune colonne de la table source n'est incluse dans la table cible. Vous pouvez utiliser <b>*NONE</b> lorsque vous voulez uniquement les colonnes calculées dans la table cible. Cette valeur est obligatoire si le paramètre <b>CALCCOL</b> contient des fonctions récapitulatives mais qu'aucun regroupement n'est effectué.</p> <p><i>nom-colonne</i> Les noms d'un maximum de 300 colonnes source que vous voulez inclure dans la table cible. Séparez les noms de colonne par des espaces.</p>

Tableau 37. Définitions des paramètres de commande ADDDPRSUBM pour System i (suite)

Paramètre	Définition et invites
<b>UNIQUE</b>	<p>Indique si la table cible a des clés uniques comme indiqué par le paramètre KEYCOL.</p> <p><b>*YES (valeur par défaut)</b>                      La table cible prend en charge une modification nette par clé ; une seule ligne existe dans la table cible pour cette clé, quel que soit le nombre de modifications effectuées sur la clé.</p> <p>Cette valeur indique que la table contient des données en cours au lieu d'un historique des modifications des données. Une table condensée comprend plus d'une ligne pour chaque clé primaire dans la table et peut être utilisée pour fournir des informations actuelles pour une régénération.</p> <p><b>*NO</b>                      La table cible prend en charge plusieurs modifications par clé. Les modifications sont jointes à la table cible.</p> <p>Cette valeur indique que la table contient un historique des modification des données au lieu des données en cours. Une table non condensée comprend plus d'une ligne pour chaque valeur clé dans la table et peut être utilisée pour fournir un historique des modifications des données. Une table non condensée ne peut pas fournir de données actuelles pour une régénération.</p>



Tableau 37. Définitions des paramètres de commande ADDDPRSUBM pour System i (suite)

Paramètre	Définition et invites
<b>KEYCOL</b>	<p>Indique les colonnes qui décrivent la clé de la table cible. les noms de colonnes doivent être non qualifiés. Pour les tables cible *POINTINTIME, *REPLICA, et *USERCOPY (comme spécifié dans le paramètre <b>TGTYPE</b>), vous devez identifier une ou plusieurs colonnes comme clé cible pour la table cible. La clé cible est utilisée par le programme Apply pour identifier chaque ligne uniquement qui change pendant la réplication des captures de modification.</p> <p><b>*SRCTBL (valeur par défaut)</b></p> <p>Les colonnes clé de la table cible sont identiques à celles de la table source. La commande <b>ADDDPRREG</b> utilise la clé spécifiée dans la table source comme si la table source avait une clé. Les colonnes clé suivantes sont utilisées :</p> <ul style="list-style-type: none"> <li>• Les colonnes clé que vous avez définies via DDS lors de la création de la table avec la commande Create Physical File (<b>CRTPF</b>)</li> <li>• Les clés primaire et unique que vous avez définies avec les instructions CREATE TABLE et ALTER TABLE SQL</li> <li>• Les clés uniques que vous avez définies avec les instructions CREATE INDEX SQL</li> </ul> <p>Si vous utilisez une colonne en tant que clé plus d'une fois et avec un ordre différent, la clé de la table cible est définie dans l'ordre croissant.</p> <p><b>*RRN</b></p> <p>La colonne clé de la table cible est la colonne IBMQSQ_RRN. La table cible est créée avec une colonne IBMQSQ_RRN, et cette colonne est utilisée comme clé. Lorsque le programme Apply s'exécute, si la table source est une table utilisateur et que la table cible est une table des points de cohérence ou une copie utilisateur, la colonne IBMQSQ_RRN dans la table cible est mise à jour avec le numéro d'enregistrement relatif de l'enregistrement associé dans la table source. Sinon, la colonne IBMQSQ_RRN dans la table cible est mise à jour avec la valeur de la colonne IBMQSQ_RRN dans la table source.</p> <p><b>*NONE</b></p> <p>La copie cible ne contient pas de clé cible. Vous ne pouvez pas spécifier *NONE si le type de la table cible est *POINTINTIME, *REPLICA, ou *USERCOPY.</p> <p><i>nom-colonne</i></p> <p>Les noms des colonnes cibles que vous voulez utiliser comme colonnes de clé cible. Vous pouvez indiquer jusqu'à 120 noms de colonnes. Séparez les noms de colonne par des espaces.</p>

Tableau 37. Définitions des paramètres de commande ADDDPRSUBM pour System i (suite)

Paramètre	Définition et invites
<b>TGTCOL</b>	<p>Indique les nouveaux noms de toutes les colonnes que le programme Apply met à jour dans la table cible. Ces noms remplacent les noms de colonne extraits de la table source. les noms de colonnes doivent être non qualifiés. Si vous avez spécifié une valeur *NONE pour le paramètre <b>COLUMN</b>, n'utilisez pas le paramètre <b>TGTCOL</b>.</p> <p>Utilisez ce paramètre pour donner des noms significatifs aux colonnes de table cible. Indiquez chaque nom de colonne source ainsi que le nom de la colonne correspondante sur la table cible.</p> <p><b>*COLUMN (valeur par défaut)</b> Les colonnes cibles sont identiques aux colonnes que vous avez spécifiées dans le paramètre <b>COLUMN</b>.</p> <p><i>nom-colonne</i> Les noms de colonne de la table source que vous voulez modifier dans la cible. Vous pouvez indiquer jusqu'à 300 noms de colonnes.</p> <p><i>nouveau-nom</i> Les nouveaux noms des colonnes cibles. Vous pouvez indiquer jusqu'à 300 nouveaux noms de colonnes. Si vous n'utilisez pas ce paramètre, le nom de la colonne dans la table cible sera identique au nom de la colonne source.</p>
<b>CALCCOL</b>	<p>Indique la liste des colonnes définies par l'utilisateur ou calculées dans la table cible. les noms de colonnes doivent être non qualifiés. Mettez chaque paire de nom de colonne et d'expression entre parenthèses.</p> <p>Vous devez spécifier un nom de colonne pour chaque expression SQL. Si vous voulez définir une colonne en tant qu'expression SQL sans clause GROUP BY, vous devez définir le paramètre <b>COLUMN</b> sur *NONE.</p> <p><b>*NONE (valeur par défaut)</b> Aucune colonne définie par l'utilisateur ou calculée n'est incluse dans la table cible.</p> <p><i>nom-colonne</i> Les noms de colonne des colonnes définies par l'utilisateur ou calculées dans la table cible. Vous pouvez indiquer jusqu'à 100 noms de colonnes.</p> <p><i>expression</i> Les expressions des colonnes définies par l'utilisateur ou calculées dans la table cible. Vous pouvez indiquer jusqu'à 100 expressions de colonnes SQL.</p>

Tableau 37. Définitions des paramètres de commande ADDDPRSUBM pour System i (suite)

Paramètre	Définition et invites
<b>ADDREG</b>	<p>Indique si la table cible est automatiquement enregistrée comme table source. Utilisez ce paramètre pour enregistrer des tables de type cible CCD.</p> <p><b>*NO (valeur par défaut)</b> La table cible n'est pas enregistrée comme table source. DB2 DataPropagator pour System i ignore cette valeur de paramètre si le type de cible est *REPLICA. Les tables cible de réplication sont toujours enregistrées en tant que tables source.</p> <p><b>*YES</b> La table cible est enregistrée en tant que table source. Cette commande échoue si vous avez déjà enregistré la table cible.</p> <p>Ne définissez pas ce paramètre sur *YES si le type de table cible est *USERCOPY, *POINTINTIME, *BASEAGR, ou *CHANGEAGR.</p> <p>Si vous définissez le paramètre <b>CRTTGTTBL</b> sur *NO, vous devez créer la table cible avant de tenter de l'enregistrer en tant que source.</p>

## Exemples pour ADDDPRSUBM

Les exemples suivants illustrent comment utiliser la commande **ADDDPRSUBM**.

### Exemple 1 :

Par exemple, pour ajouter un membre d'ensemble d'abonnements à l'ensemble d'abonnements SETHR sous le qualificatif Apply AQHR :

```
ADDDPRSUBM APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/YTD TAX) TGTTBL(TGTHR/TGTTAX)
```

### Exemple 2 :

Pour ajouter un membre d'un ensemble d'abonnements avec seulement deux colonnes, AMOUNT et NAME, depuis la table source enregistrée nommée YTD TAX et répliquer ces colonnes dans une table cible existante nommée TGTTAX :

```
ADDDPRSUBM APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/YTD TAX) TGTTBL(TGTLIB/TGTTAX)
  CRTTGTTBL(*NO) COLUMN(AMOUNT NAME) CHKfmt(*YES)
```

Cette commande vérifie que les colonnes AMOUNT et NAME définies pour ce membre de l'ensemble d'abonnements correspondent aux colonnes de la table cible.

### Exemple 3 :

Pour ajouter un membre d'ensemble d'abonnements à l'ensemble d'abonnements nommé SETHR et répliquer ces données dans une table cible de modification cohérente des données nommée TGTYTD :

```
ADDDPRSUBM APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/YTD TAX) TGTTBL(TGTLIB/TGTYTD)
  TGTTYpe(*CCD) ADDREG (*YES)
```

Cette commande enregistre la table cible en tant que table source pour DB2 DataPropagator pour System i.

## ANZDPR : fonctionnement de l'Analyseur (System i)

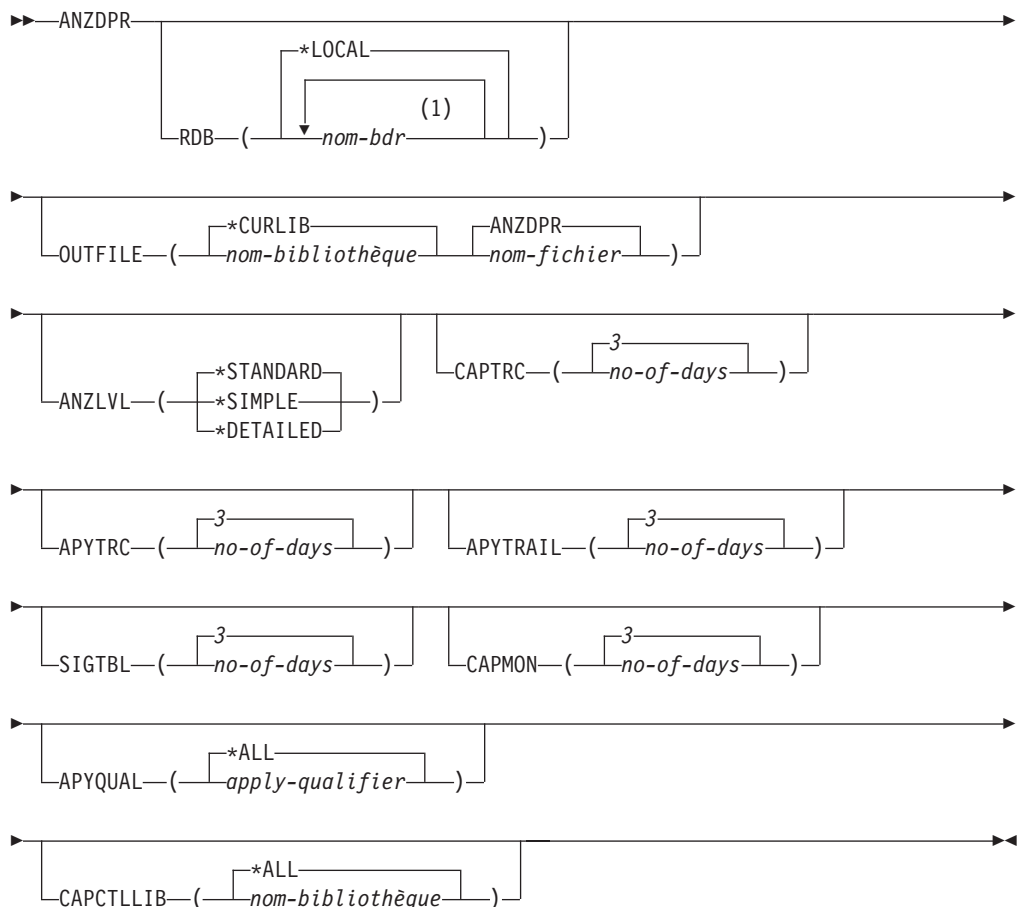
Utilisez la commande Analyze DPR (**ANZDPR**) pour analyser un échec du programme Capture ou Apply, pour vérifier le paramétrage de votre configuration de réplication, ou pour obtenir un diagnostic d'incident et des informations sur le réglage des performances.

Exécutez cette commande après avoir paramétré votre configuration de réplication.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe



### Remarques :

1 Vous pouvez spécifier jusqu'à 10 bases de données.

Le tableau 38, à la page 353 dresse la liste des paramètres d'appel.

Tableau 38. Définitions des paramètres de commande ANZDPR pour System i

Paramètre	Définition et invites
<b>RDB</b>	<p>Indique les bases de données à analyser.</p> <p><b>*LOCAL (par défaut)</b> La base de données de votre système local.</p> <p><i>nom-bdr</i> Le nom d'entrée de répertoire RDB, qui indique la base de données.</p> <p>Vous pouvez spécifier jusqu'à 10 bases de données. Si vous voulez analyser plusieurs bases de données, y compris la base de données sur votre système local, veillez à ce que *LOCAL soit la première entrée de la liste. De même, vérifiez que vous vous connectez à toutes ces bases de données à partir de votre système actuel.</p>
<b>OUTFILE</b>	<p>Indique la bibliothèque et le nom de fichier utilisés pour stocker la sortie d'analyse. Cette commande écrit la sortie sur un fichier HTML.</p> <p><b>*CURLIB (valeur par défaut)</b> La bibliothèque en cours.</p> <p><i>nom-bibliothèque</i> Le nom de la bibliothèque.</p> <p><b>ANZDPR (valeur par défaut)</b> La sortie est écrite dans un fichier HTML nommé ANZDPR.</p> <p><i>nom-fichier</i> Le nom du fichier de sortie HTML.</p> <p>Si le nom de fichier existe déjà, le fichier est écrasé. Si le nom de fichier n'existe pas, la commande crée le fichier avec les attributs RCDLEN(512) et SIZE(*NOMAX).</p>
<b>ANZLVL</b>	<p>Indique le niveau d'analyse à signaler. Le niveau d'analyse peut être :</p> <p><b>*STANDARD (valeur par défaut)</b> Génère un rapport incluant le contenu des tables de contrôle et les informations de statut des programmes Capture et Apply.</p> <p><b>*SIMPLE</b> Génère les informations dans le rapport standard mais exclut les détails des sous-colonnes. Utilisez cette option si vous voulez générer un rapport plus petit qui nécessite moins de ressources système.</p> <p><b>*DETAILED</b> Génère un rapport avec l'analyse la plus complète. Le rapport détaillé comprend des informations du rapport standard en plus des informations sur l'ensemble d'abonnements.</p>
<b>CAPTRC</b>	<p>Indique l'intervalle de dates (0 à 30 jours) des entrées devant faire l'objet d'un rapport de la table IBMSNAP_CAPTRACE. La valeur par défaut est de 3.</p> <p><i>no-of-days</i> Le nombre de jours faisant l'objet du rapport.</p>
<b>APYTRC</b>	<p>Indique l'intervalle de dates (0 à 30 jours) des entrées devant faire l'objet d'un rapport de la table IBMSNAP_APPLYTRACE. La valeur par défaut est de 3.</p> <p><i>no-of-days</i> Le nombre de jours faisant l'objet du rapport.</p>

Tableau 38. Définitions des paramètres de commande ANZDPR pour System i (suite)

Paramètre	Définition et invites
<b>APYTRAIL</b>	Indique l'intervalle de date (de 0 à 30 jours) des entrées à extraire de la table IBMSNAP_APPLYTRAIL. La valeur par défaut est de 3.  <i>no-of-days</i> Le nombre de jours faisant l'objet du rapport.
<b>SIGTBL</b>	Indique l'intervalle de dates (0 à 30 jours) des entrées devant faire l'objet d'un rapport de la table IBMSNAP_SIGNAL. La valeur par défaut est de 3.  <i>no-of-days</i> Le nombre de jours faisant l'objet du rapport.
<b>CAPMON</b>	Indique l'intervalle de dates (0 à 30 jours) des entrées devant faire l'objet d'un rapport de la table IBMSNAP_CAPMON. La valeur par défaut est de 3.  <i>no-of-days</i> Le nombre de jours faisant l'objet du rapport.
<b>APYQUAL</b>	Indique les qualificatifs Apply à analyser.  <b>*ALL (valeur par défaut)</b> Tous les qualificatifs Apply sont analysés.  <i>apply-qualifier</i> Nom du qualificatif Apply à analyser. Vous pouvez spécifier jusqu'à 10 qualificatifs Apply.
<b>CAPCTLLIB</b>	Indique les schémas Capture, qui correspondent aux noms des bibliothèques de contrôle Capture que vous voulez analyser. Vous pouvez analyser une bibliothèque de contrôle Capture spécifique, ou vous pouvez choisir la valeur par défaut *ALL pour analyser toutes les bibliothèques de contrôle Capture.  <b>*ALL(valeur par défaut)</b> Toutes les bibliothèques de contrôle Capture seront analysées.  <i>nom-bibliothèque</i> Le nom de la bibliothèque de contrôle Capture spécifique que vous voulez analyser.

## Exemples pour ANZDPR

Les exemples suivants illustrent comment utiliser la commande **ANZDPR**.

### Exemple 1 :

Pour exécuter l'Analyseur sur votre base de données locale et sur une base de données distante appelée RMTRDB1 en utilisant un niveau d'analyse standard :

```
ANZDPR RDB(*LOCAL RMTRDB1) OUTFILE(MYLIB/ANZDPR) ANZLVL(*STANDARD) CAPTRC(1)
  APYTRC(1) APYTRAIL(1) SIGTBL(1) CAPMON(1) APYQUAL(*ALL)
```

Cet exemple génère une journée d'entrées des tables IBMSNAP\_CAPTRACE, IBMSNAP\_APPLYTRACE, IBMSNAP\_APPLYTRAIL, IBMSNAP\_SIGNAL, et IBMSNAP\_CAPMON pour tous les qualificatifs Apply et écrit la sortie dans un fichier HTML nommé ANZDPR situé dans la bibliothèque nommée MYLIB.

### Exemple 2 :

Pour exécuter l'Analyseur avec toutes les valeurs par défaut :

## CHGDPRCAPA : modification des attributs DPR Capture (System i)

Utilisez la commande Change DPR Capture Attributes (**CHGDPRCAPA**) pour modifier les paramètres d'exploitation globaux utilisés par le programme Capture et stockés dans la table IBMSNAP\_CAPPARMS.

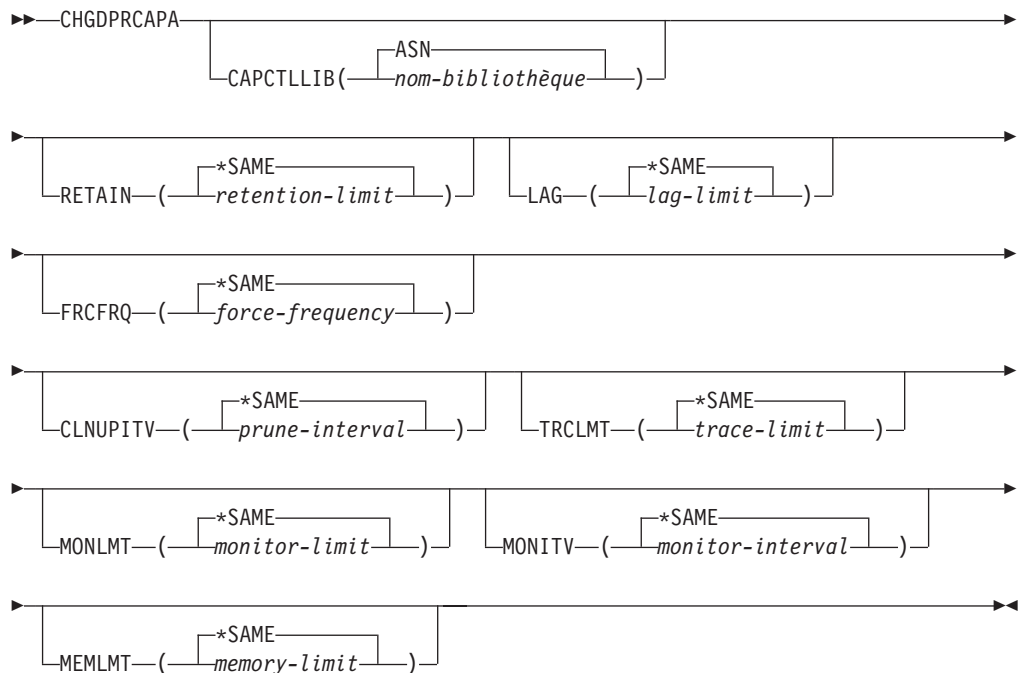
Ces modifications de paramètres ne prennent pas effet tant que vous n'avez pas effectué l'une des actions suivantes :

- Emettre une commande **INZDPRCAP**.
- Arrêter puis relancer le programme Capture.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe



Le tableau 39, à la page 356 dresse la liste des paramètres d'appel.

Tableau 39. Définitions des paramètres de commande CHGDPRCAPA pour System i

Paramètre	Définition et invites
<b>CAPCTLLIB</b>	<p>Spécifie le schéma Capture, qui est le nom de la bibliothèque dans laquelle résident les tables de contrôle Capture.</p> <p><b>ASN (valeur par défaut)</b> Les tables de contrôle Capture se trouvent dans la bibliothèque ASN.</p> <p><i>nom-bibliothèque</i> Nom d'une bibliothèque contenant les tables de contrôle de Capture.</p>
<b>RETAIN</b>	<p>Indique la nouvelle durée de conservation, qui correspond au nombre de minutes pendant lesquelles les données sont conservées dans la table de modification des données (CD), la table des unités d'oeuvre (UOW), les tables IBMSNAP_SIGNAL, et IBMSNAP_AUTHTKN avant d'être supprimées. Cette valeur est stockée dans la colonne RETENTION_LIMIT de la table IBMSNAP_CAPPARMS.</p> <p>Cette valeur fonctionne avec la valeur de paramètre <b>CLNUPITV</b>. Lorsque la valeur <b>CLNUPITV</b> est atteinte, des données CD, UOW, IBMSNAP_SIGNAL, et IBMSNAP_AUTHTKN sont supprimées si ces données dépassent la limite de conservation.</p> <p>Veillez à ce que les intervalles Apply soient définis afin de copier les informations modifiées avant que les données n'atteignent cette valeur de paramètre <b>RETAIN</b>, et ce, afin d'éviter des données incohérentes dans vos tables. Si les données deviennent incohérentes, le programme Apply effectue une régénération intégrale.</p> <p>La valeur par défaut est de 10 080 minutes (sept jours). Le maximum est de 35000000 minutes.</p> <p><b>*SAME (valeur par défaut)</b> Cette valeur n'est pas modifiée.</p> <p><i>retention-limit</i> La nouvelle valeur de durée de conservation.</p>
<b>LAG</b>	<p>Spécifie la nouvelle limite de décalage, qui correspond au nombre de minutes pendant lesquelles le programme Capture peut subir un retard de traitement avant de redémarrer. Cette valeur est stockée dans la colonne LAG_LIMIT de la table IBMSNAP_CAPPARMS.</p> <p>Lorsque la limite de décalage est atteinte (à savoir, lorsque l'horodatage de l'entrée de journal est antérieur à l'horodatage actuel moins la limite de décalage), le programme Capture lance un démarrage à froid pour les tables qu'il traite dans ce journal. Le programme Apply effectue ensuite une régénération intégrale afin de fournir au programme Capture un nouveau point de départ.</p> <p>La valeur par défaut est de 10 080 minutes (sept jours). Le maximum est de 35000000 minutes.</p> <p><b>*SAME (valeur par défaut)</b> Cette valeur n'est pas modifiée.</p> <p><i>lag-limit</i> La nouvelle valeur de limite de décalage.</p>



Tableau 39. Définitions des paramètres de commande CHGDPRCAPA pour System i (suite)

Paramètre	Définition et invites
<b>FRCFRQ</b>	<p>Indique la fréquence (de 30 à 600 secondes) selon laquelle le programme Capture écrit les modifications dans les tables de données de modification (CD) et d'unités d'oeuvre (UOW). Cette valeur est stockée dans la colonne COMMIT_INTERVAL de la table IBMSNAP_CAPPARMS.</p> <p>Le programme Capture met ces modifications à la disposition du programme Apply soit lorsque les mémoires tampons sont remplies, soit lorsque la limite de temps <b>FRCFRQ</b> expire, quel que soit le plus tôt.</p> <p>Utilisez ce paramètre pour rendre les modifications immédiatement disponibles pour le programme Apply sur les serveurs avec un faible taux de modifications de table source. La valeur de paramètre <b>FRCFRQ</b> est une valeur globale utilisée pour toutes les tables sources définies. La configuration de la valeur <b>FRCFRQ</b> sur un faible nombre peut affecter les performances du système.</p> <p>La valeur par défaut est de 30 secondes.</p> <p><b>*SAME (valeur par défaut)</b>            Cette valeur n'est pas modifiée.</p> <p><i>force-frequency</i>            La nouvelle valeur d'intervalle de validation, qui correspond au nombre de secondes pendant lesquelles le programme Capture conserve les modifications de la table de données de modification et des unités d'oeuvre dans l'espace de la mémoire tampon avant de mettre ces modifications à la disposition du programme Apply.</p>
<b>CLNUPITV</b>	<p>Indique la durée maximale (en heures) avant que le programme Capture n'élague les anciens enregistrements des tables de modification de données (CD), des tables d'unités d'oeuvre (UOW), et des tables IBMSNAP_SIGNAL, IBMSNAP_CAPMON, IBMSNAP_CAPTRACE, et IBMSNAP_AUTHTKN.</p> <p>Ce paramètre fonctionne en conjonction avec le paramètre <b>RETAIN</b> pour contrôler l'élagage des tables CD, UOW, IBMSNAP_SIGNAL, et IBMSNAP_AUTHTKN, avec le paramètre <b>MONLMT</b> pour contrôler l'élagage de la table IBMSNAP_CAPMON, et avec le paramètre <b>TRCLMT</b> pour contrôler l'élagage de la table IBMSNAP_CAPTRACE. (Utilisez la commande <b>STRDPRCAP</b> pour définir les paramètres <b>RETAIN</b>, <b>MONLMT</b>, et <b>TRCLMT</b> pour un programme Capture.</p> <p>La valeur de ce paramètre est automatiquement convertie d'heures en secondes et est stockée dans la colonne PRUNE_INTERVAL de la table IBMSNAP_CAPPARMS. Si la colonne PRUNE_INTERVAL est modifiée manuellement (sans l'aide de la commande <b>CHGDPRCAPA</b>), des modifications dues à l'arrondi peuvent apparaître lorsque vous faites une invite avec la touche F4.</p> <p><b>*SAME (valeur par défaut)</b>            Cette valeur d'attribut Capture n'est pas modifiée.</p> <p><i>prune-interval</i>            L'intervalle d'élagage exprimé en nombre d'heures spécifique (1 à 100).</p>

Tableau 39. Définitions des paramètres de commande CHGDPRCAPA pour System i (suite)

Paramètre	Définition et invites
TRCLMT	<p>Spécifie la limite de trace (en minutes). Cette valeur est stockée dans la colonne TRACE_LIMIT de la table IBMSNAP_CAPPARMS.</p> <p>Le programme Capture élague toute ligne de table IBMSNAP_CAPTRACE qui est plus ancienne que la limite de trace. La valeur par défaut est 10 080 minutes (sept jours d'entrées de trace).</p> <p><b>*SAME (valeur par défaut)</b> Cette valeur n'est pas modifiée.</p> <p><i>trace-limit</i> Le nombre de minutes de données de trace conservées dans la table IBMSNAP_CAPTRACE après l'élagage.</p>
MONLMT	<p>Spécifie le nombre maximal de contrôles (en minutes). Cette valeur est stockée dans la colonne MONITOR_LIMIT de la table IBMSNAP_CAPPARMS.</p> <p>Le programme Capture élague toute ligne de la table IBMSNAP_CAPMON qui est plus ancienne que le nombre maximal de contrôles.</p> <p>La valeur par défaut est 10 080 minutes (sept jours d'entrées de contrôles).</p> <p><b>*SAME (valeur par défaut)</b> Cette valeur n'est pas modifiée.</p> <p><i>monitor-limit</i> Le nombre de minutes de données de contrôle conservées dans la table IBMSNAP_CAPMON après l'élagage.</p>
MONITV	<p>Indique à quelle fréquence (en secondes) le programme Capture insère des lignes dans la table IBMSNAP_CAPMON. Cette valeur est stockée dans la colonne MONITOR_INTERVAL de la table IBMSNAP_CAPPARMS.</p> <p>La valeur par défaut est de 300 secondes (cinq minutes).</p> <p><b>*SAME (valeur par défaut)</b> Cette valeur n'est pas modifiée.</p> <p><i>monitor-interval</i> Le nombre de secondes entre l'insertion de ligne dans la table IBMSNAP_CAPMON. L'intervalle de contrôle doit être d'au moins 120 secondes (deux minutes). Si vous saisissez un nombre inférieur à 120, cette commande définit automatiquement cette valeur de paramètre sur 120.</p>
MEMLMT	<p>Spécifie la taille maximale (en mégaoctets) de mémoire que le travail du journal Capture peut utiliser. Cette valeur est stockée dans la colonne MEMORY_LIMIT de la table IBMSNAP_CAPPARMS.</p> <p>La valeur par défaut est de 32 mégaoctets.</p> <p><b>*SAME (valeur par défaut)</b> Cette valeur n'est pas modifiée.</p> <p><i>memory-limit</i> Le nombre maximal de mégaoctets pour la mémoire.</p>



Le tableau 40 dresse la liste des paramètres d'appel.

Tableau 40. Définitions des paramètres de commande CRTDPRTBL pour System i

Paramètre	Définition et invites
<b>CAPCTLLIB</b>	Spécifie le schéma Capture, qui est le nom de la bibliothèque dans laquelle sont placées les tables de contrôle Capture récemment créées.  <b>ASN (valeur par défaut)</b> Les tables de contrôle Capture sont placées dans la bibliothèque ASN.  <i>nom-bibliothèque</i> Nom de la bibliothèque dans laquelle sont placées les tables de contrôle Capture.

## Exemples pour CRTDPRTBL

Les exemples suivants illustrent comment utiliser la commande **CRTDPRTBL**.

### Exemple 1 :

Pour créer de nouvelles tables de contrôle de réplication dans la bibliothèque ASN par défaut :

```
CRTDPRTBL CAPCTLLIB(ASN)
```

### Exemple 2 :

Pour créer de nouvelles tables de contrôle de réplication pour un schéma Capture appelé DPRSALES :

```
CRTDPRTBL CAPCTLLIB(DPRSALES)
```

## ENDDPRAPY : arrêt de Apply (System i)

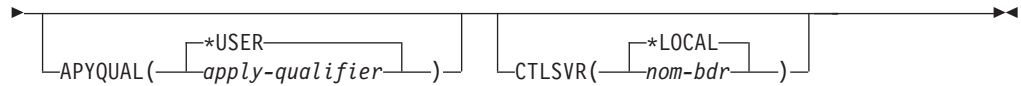
Utilisez la commande End DPR Apply (**ENDDPRAPY**) pour arrêter un programme Apply sur votre système local.

Vous devez arrêter le programme Apply avant tout arrêt planifié du système. Il se peut que vous souhaitiez arrêter le programme Apply pendant des périodes de pic d'utilisation du système.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

```
▶▶ ENDDPRAPY _____
                    |_____|
                    |_____| *CURRENT _____|
                    |_____| USER(_____|user-name|_____|)
                    |_____| OPTION(_____|*CNTRLD
                    |_____| *IMMED _____)|
```



Le tableau 41 dresse la liste des paramètres d'appel.

Tableau 41. Définitions des paramètres de commande ENDDPRAPY pour System i

Paramètre	Définition et invites
<b>USER</b>	<p>Ce paramètre est ignoré, sauf si le paramètre <b>APYQUAL</b> a la valeur <b>*USER</b>, auquel cas, ce paramètre spécifie le qualificatif Apply associé au programme Apply.</p> <p><b>*CURRENT (valeur par défaut)</b>            Le programme Apply de l'utilisateur associé au travail en cours.</p> <p><i>user-name</i>            Le programme Apply de l'utilisateur spécifié.</p> <p>Lors de l'invite sur la commande <b>ENDDPRAPY</b>, vous pouvez appuyer sur la touche F4 pour afficher une liste des utilisateurs qui ont défini des ensembles d'abonnements.</p>
<b>OPTION</b>	<p>Indique comment arrêter le programme Apply.</p> <p><b>*CNTRLD (valeur par défaut)</b>            Le programme Apply termine toutes ses tâches avant de s'arrêter. Ces tâches peuvent prendre beaucoup de temps si le programme Apply termine un ensemble d'abonnements.</p> <p><b>*IMMED</b>            Le programme Apply termine toutes ses tâches avec la commande <b>ENDJOB OPTION(*IMMED)</b>. Les tâches se terminent immédiatement, sans nettoyage. Utilisez cette option uniquement après l'échec d'une fin contrôlée, car elle peut provoquer des résultats indésirables. (Sauf si le programme Apply était en veille lorsque vous avez émis la commande <b>ENDDPRAPY</b>, vous devez vérifier le contenu de la table cible.)</p> <p>Si le programme Apply effectuait une régénération intégrale dans la table cible, la table cible peut être vide à la suite de l'arrêt du programme Apply avant que la table ait été régénérée avec le contenu de la table source. Si la table cible est vide, vous devez forcer une régénération complète pour cette cible de réplication.</p> <p>Vous pouvez découvrir qu'un ensemble d'abonnements est considéré IN USE (la colonne STATUS dans la table IBMSNAP_SUBS_SET a une valeur de 1). Si tel est le cas, réinitialisez la valeur à 0 ou -1. Cela permet au programme Apply d'exécuter de nouveau l'ensemble d'abonnements.</p>

Tableau 41. Définitions des paramètres de commande ENDDPRAPY pour System i (suite)

Paramètre	Définition et invites
APYQUAL	<p>Indique le qualificatif Apply utilisé par le programme Apply.</p> <p><b>*USER (valeur par défaut)</b> Le nom d'utilisateur spécifié dans le paramètre <b>USER</b> est le qualificatif Apply.</p> <p><i>apply-qualifier</i> Le nom utilisé pour regrouper les ensembles d'abonnements qui doivent être exécutés par ce programme Apply. Pour le nom du qualificatif Apply, vous pouvez spécifier un maximum de 18 caractères. Ce nom suit les mêmes conventions de dénomination qu'un nom de base de données relationnelle. Vous identifiez les abonnements exécutés par les enregistrements de la table IBMSNAP_SUBS_SET ayant cette valeur dans la colonne APPLY_QUAL.</p> <p>Lors de l'invite sur la commande <b>ENDDPRAPY</b>, vous pouvez appuyer sur la touche F4 pour afficher une liste des noms de qualificatifs Apply avec des ensembles d'abonnements existants.</p>
CTLSVR	<p>Spécifie le nom de la base de données relationnelle du système contenant les tables de contrôle Apply.</p> <p><b>*LOCAL (par défaut)</b> Les tables de contrôle Apply résident localement (sur l'ordinateur sur lequel vous exécutez la commande <b>ENDDPRAPY</b>).</p> <p><i>nom-bdr</i> Le nom de la base de données relationnelle où résident les tables de contrôle Apply. Vous pouvez utiliser la commande Work with RDB Directory Entries (<b>WRKRDBDIRE</b>) pour trouver ce nom.</p> <p>Lors de l'invite sur la commande <b>ENDDPRAPY</b>, vous pouvez appuyer sur la touche F4 pour afficher une liste des noms RDB disponibles.</p>

## Notes sur l'utilisation

La commande **ENDDPRAPY** utilise la valeur des paramètres **APYQUAL** et **CTLSVR** pour rechercher dans la table IBMSNAP\_APPLY\_JOB le nom du travail, le numéro du travail, et l'utilisateur du travail pour le programme Apply référencé, et termine ce travail.

**ENDDPRAPY** émet un message d'erreur si les conditions suivantes se produisent :

- Si la table IBMSNAP\_APPLY\_JOB ne quitte pas ou est corrompue.
- S'il n'y a pas d'enregistrement dans la table IBMSNAP\_APPLY\_JOB pour le qualificatif Apply et le nom du serveur de contrôle.
- Si le travail Apply est déjà terminé.
- Si l'ID utilisateur exécutant la commande n'est pas autorisé à terminer le travail Apply.

## Exemples pour ENDDPRAPY

Les exemples suivants illustrent comment utiliser la commande **ENDDPRAPY**.

### Exemple 1 :

Pour arrêter le programme Apply qui utilise le qualificatif Apply AQHR :

```
ENDDPRAPY OPTION(*CNTRLD) APYQUAL(AQHR)
```

Le programme Apply s'arrête dès que toutes se tâches sont terminées.

### Exemple 2 :

Pour arrêter immédiatement le programme Apply :

```
ENDDPRAPY OPTION(*IMMED) APYQUAL(AQHR)
```

Les tâches du programme Apply se terminent immédiatement, sans nettoyage.

### Exemple 3 :

Arrêter un programme Apply qui utilise les tables de contrôle Apply qui résident dans une base de données relationnelle nommée DB1X :

```
ENDDPRAPY OPTION(*CNTRLD) APYQUAL(AQHR) CTLSVR(DB1X)
```

---

## ENDDPRCAP : arrêt de Capture (System i)

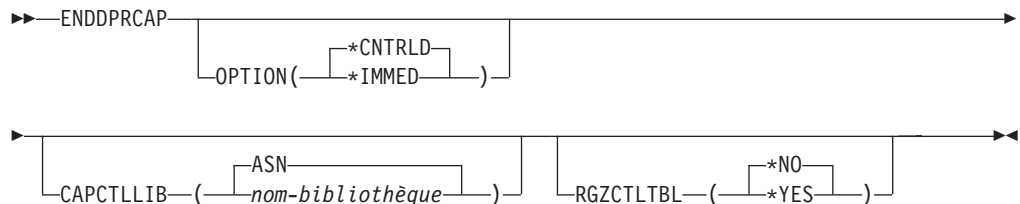
Utilisez la commande End DPR Capture (**ENDDPRCAP**) pour arrêter le programme Capture.

Utilisez cette commande pour arrêter le programme Capture avant d'arrêter le système. Il se peut que vous souhaitiez arrêter le programme pendant des périodes de pic d'utilisation du système afin d'augmenter les performances des autres programmes qui s'exécutent sur le système.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe



Le tableau 42, à la page 364 dresse la liste des paramètres d'appel.

Tableau 42. Définitions des paramètres de commande **ENDDPRCAP** pour System i

Paramètre	Définition et invites
<b>OPTION</b>	<p>Indique comment arrêter le programme Capture.</p> <p><b>*CNTRLD (valeur par défaut)</b> Le programme Capture s'arrête normalement après l'exécution de toutes les tâches.</p> <p>La commande <b>ENDDPRCAP</b> peut prendre plus de temps lorsque vous spécifiez l'option <b>*CNTRLD</b> car le programme Capture exécute tous ses processus subordonnés avant de s'arrêter.</p> <p><b>*IMMED</b> Le programme Capture s'arrête normalement après avoir terminé toutes les tâches avec la commande <b>ENDJOB OPTION(*IMMED)</b>.</p>
<b>CAPCTLLIB</b>	<p>Spécifie le schéma Capture, qui est le nom de la bibliothèque dans laquelle se trouvent les tables de contrôle Capture. Cette bibliothèque comprend la table <b>IBMSNAP_REGISTER</b>, qui stocke les informations d'enregistrement des tables source.</p> <p><b>ASN (valeur par défaut)</b> les tables de contrôle Capture se trouvent dans la bibliothèque ASN. La bibliothèque ASN est la bibliothèque par défaut.</p> <p><i>nom-bibliothèque</i> Nom d'une bibliothèque contenant les tables de contrôle de Capture.</p>
<b>RGZCTLBTL</b>	<p>Indique si une commande Reorganize Physical File Member (<b>RGZPFM</b>) est exécutée sur les tables de contrôle (y compris les tables de données de modification (CD) et d'unités d'oeuvre(UOW)) lorsque se termine le programme Capture. Le système ne récupère pas d'espace disque sauf si le processus de commande <b>RGZPFM</b> est exécuté dans les tables. La commande <b>RGZPFM</b> ne sera pas exécutée si le programme Apply ou d'autres programmes d'application accèdent aux tables de contrôle.</p> <p><b>*NO (valeur par défaut)</b> La commande <b>RGZPFM</b> n'est pas exécutée.</p> <p><b>*YES</b> La commande <b>RGZPFM</b> est exécutée.</p>

## Notes sur l'utilisation

Si vous utilisez la commande **ENDJOB**, des objets temporaires pourraient être laissés dans la bibliothèque QDP4. Ces objets ont les types **\*DTAQ** et **\*USRSPC**, et sont nommés **QDP4nnnnnn**, où **nnnnnn** est le numéro de travail du travail qui les a utilisés. Vous pouvez supprimer ces objets lorsque le travail qui les a utilisés (identifiés par le numéro de travail dans le nom d'objet) n'est pas actif.

Si le travail sous la bibliothèque de contrôle Capture ne se termine pas après l'émission de cette commande, utilisez la commande **ENDJOB** avec l'option **\*IMMED** afin de terminer ce travail et tous les travaux du journal s'exécutant dans le sous-système DB2 DataPropagator pour System i. Ne terminez pas les travaux Apply s'exécutant dans le même sous-système si vous voulez terminer uniquement le programme Capture.

Dans les rares cas où le travail de contrôle Capture se termine anormalement, les travaux de journal créés par le travail de contrôle de Capture (qui est nommé conformément au paramètre **CAPCTLLIB**) pourraient encore être en cours



d'exécution. La seule manière de terminer des travaux est d'utiliser la commande **ENDJOB** avec l'option \*IMMED ou \*CNTRLD.

## Exemples pour ENDDPRCAP

Les exemples suivants illustrent la manière d'utiliser la commande **ENDDPRCAP**.

### Exemple 1 :

Pour terminer le programme Capture, qui utilise les tables de contrôle Capture dans la bibliothèque ASN, après que toutes les tâches de traitement soient terminées :

```
ENDDPRCAP OPTION(*CNTRLD) CAPCTLLIB(ASN) RGZCTLTBL(*NO)
```

### Exemple 2 :

Pour terminer immédiatement le programme Capture pour le schéma de Capture BSN :

```
ENDDPRCAP OPTION(*IMMED) CAPCTLLIB(BSN) RGZCTLTBL(*NO)
```

### Exemple 3 :

Pour terminer le programme Capture après que toutes les tâches de traitement soient terminées et afin de réorganiser les tables de contrôle de Capture :

```
ENDDPRCAP OPTION(*CNTRLD) CAPCTLLIB(ASN) RGZCTLTBL(*YES)
```

---

## GRTDPRAUT : autorisation des utilisateurs (System i)

Utilisez la commande Grant DPR Authority (**GRTDPRAUT**) pour autoriser une liste d'utilisateurs à accéder aux tables de contrôle de réplication afin d'exécuter les programmes Capture et Apply.

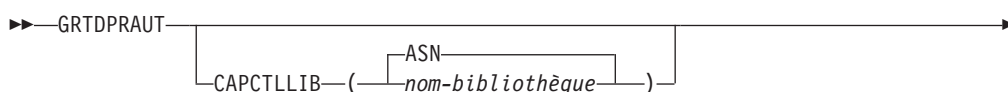
Par exemple, les exigences de droits d'accès pour l'utilisateur qui exécute les programmes Capture et Apply peuvent varier des exigences de droits d'accès pour l'utilisateur qui définit des sources et cibles de réplication.

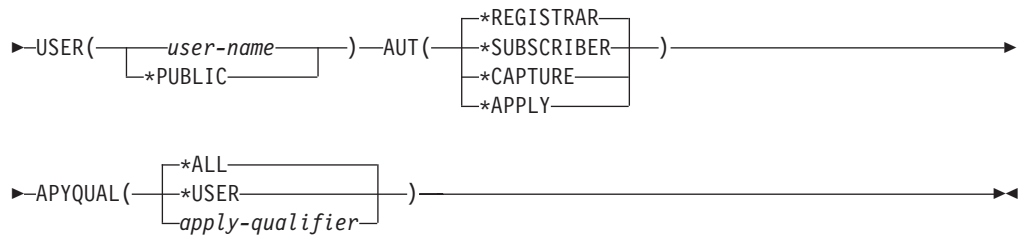
Vous devez bénéficier de l'autorité \*ALLOBJ pour accorder des droits d'accès.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe





Le tableau 43 dresse la liste des paramètres d'appel.

Tableau 43. Définitions des paramètres de commande GRTDPRAUT pour System i

Paramètre	Définition et invites
<b>CAPCTLLIB</b>	<p>Spécifie le schéma Capture, qui est le nom de la bibliothèque qui contient les tables de contrôle de réplication auxquelles l'utilisateur se voit accorder l'accès.</p> <p><b>ASN (valeur par défaut)</b> Les tables de contrôle Capture se trouvent dans la bibliothèque ASN.</p> <p><i>nom-bibliothèque</i> Nom de la bibliothèque contenant les tables de contrôle de réplication.</p>
<b>USER</b>	<p>Indique les utilisateurs qui bénéficient des droits d'accès.</p> <p><i>user-name</i> Indique un nombre maximal de 50 utilisateurs qui bénéficient des droits d'accès.</p> <p><b>*PUBLIC</b> Indique que le droit d'accès *PUBLIC est accordé sur le fichier, mais (s'il est insuffisant pour la tâche) il est utilisé uniquement pour les utilisateurs qui n'ont pas de droits d'accès spécifiques, qui ne sont pas sur la liste d'autorisation associée au fichier, et dont le profil de groupe n'a pas de droits d'accès.</p>

Tableau 43. Définitions des paramètres de commande GRTDPRAUT pour System i (suite)

Paramètre	Définition et invites
AUT	<p>Indique le type de droits d'accès accordé.</p> <p><b>*REGISTRAR (valeur par défaut)</b>            Les utilisateurs reçoivent les droits d'accès pour définir, modifier et supprimer les enregistrements.</p> <p>Pour une liste complète des droits d'accès avec AUT(*REGISTRAR), voir le tableau 44, à la page 368.</p> <p><b>*SUBSCRIBER</b>            Les utilisateurs reçoivent les droits d'accès pour définir, modifier et supprimer les ensembles d'abonnements.</p> <p>Pour une liste complète des droits d'accès avec AUT(*SUBSCRIBER), voir le tableau 45, à la page 369.</p> <p><b>*CAPTURE</b>            Les utilisateurs reçoivent les droits d'accès pour exécuter le programme Capture.</p> <p>Pour une liste complète des droits d'accès avec AUT(*CAPTURE), voir le tableau 46, à la page 370.</p> <p><b>*APPLY</b>            Les utilisateurs reçoivent les droits d'accès pour exécuter le programme Apply.</p> <p>La commande ne concède pas de droits d'accès à l'un des objets résidant sur d'autres bases de données auxquelles accède le programme Apply.</p> <p>Lorsqu'un programme Apply est appelé; l'utilisateur associé au travail du serveur d'application DRDA doit également bénéficier du droit d'accès *APPLY. Si la source est un serveur System i, vous devez exécuter la commande <b>GRTDPRAUT</b> sur le système du serveur source, en indiquant l'utilisateur du travail du serveur d'application dans le paramètre <b>USER</b> et en indiquant le qualificatif Apply dans le paramètre <b>APYQUAL</b>.</p> <p>Les droits d'accès ne sont pas accordés sur les tables cible sauf si le serveur cible est le même que le serveur de contrôle et que tous deux résident sur le système sur lequel la commande est exécutée.</p> <p>Pour une liste complète des droits d'accès accordés avec AUT(*APPLY), voir le tableau 47, à la page 372.</p>

Tableau 43. Définitions des paramètres de commande **GRTDPRAUT** pour System i (suite)

Paramètre	Définition et invites
<b>APYQUAL</b>	<p>Indique le qualificatif Apply devant être utilisé par le programme Apply avec le paramètre <b>USER</b>. Ce paramètre est utilisé uniquement lorsque <b>AUT(*APPLY)</b> ou <b>AUT(*SUBSCRIBER)</b> est spécifié.</p> <p><b>*ALL (valeur par défaut)</b> L'utilisateur reçoit le droit d'accès pour exécuter le programme Apply ou pour définir et supprimer les ensembles d'abonnements pour tous les qualificatifs Apply.</p> <p><b>*USER</b> Les utilisateurs spécifiés dans le paramètre <b>USER</b> bénéficient du droit d'accès aux ensembles d'abonnements avec un qualificatif Apply qui est identique au nom d'utilisateur.</p> <p><i>apply-qualifier</i> L'utilisateur reçoit le droit d'accès pour exécuter le programme Apply ou pour définir et supprimer les ensembles d'abonnements pour les qualificatifs Apply associés à ce qualificatif Apply.</p> <ul style="list-style-type: none"> <li>L'utilisateur reçoit le droit d'accès à toutes les sources de réplication, les tables de données de modification (CD), et les tables de modification cohérente des données (CCD) associées aux enregistrements dans la table <b>IBMSNAP_PRUNCNTL</b> et dont la valeur dans la colonne <b>APPLY_QUAL</b> correspond à la saisie de valeur avec le paramètre <b>APYQUAL</b>.</li> <li>L'utilisateur reçoit le droit d'accès aux ensembles d'abonnements listés dans la table <b>IBMSNAP_SUBS_MEMBR</b> qui résident sur ce système.</li> </ul>

## Notes sur l'utilisation

Vous ne pouvez pas utiliser la commande **GRTDPRAUT** pendant que les programmes Capture ou Apply s'exécutent, ou lorsque les application qui utilisent les tables source sont actives car les autorisations ne peuvent pas être modifiées sur des fichiers en cours d'utilisation.

Les tables suivantes dressent la liste des droits d'accès accordés lorsque vous indiquez :

- **AUT(\*REGISTRAR)**
- **AUT\*(SUBSCRIBER)**
- **AUT(\*CAPTURE)**
- **AUT(\*APPLY)**

dans la commande **GRTDPRAUT**.

La table suivante dresse la liste des droits d'accès accordés lorsque vous indiquez le paramètre **AUT(\*REGISTRAR)** dans la commande **GRTDPRAUT**.

Tableau 44. Droits d'accès accordés avec **GRTDPRAUT AUT(\*REGISTRAR)**

Bibliothèque	Objet	Type	Droits d'accès
QSYS	capctllib	*LIB	*USE, *ADD
capctllib <sup>1</sup>	QSQRJN	*JRN	*OBJOPR, *OBJMGT

Tableau 44. Droits d'accès accordés avec GRTDPRAUT AUT(\*REGISTRAR) (suite)

Bibliothèque	Objet	Type	Droits d'accès
capctllib <sup>1</sup>	QZS8CTLBLK	*USRSPC	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_REGISTER	*FILE	*OBJOPR, *READ, *ADD, *UPDT, *DLT
capctllib <sup>1</sup>	IBMSNAP_REGISTERX	*FILE	*OBJOPR, *READ, *ADD, *UPDT, *DLT
capctllib <sup>1</sup>	IBMSNAP_REGISTERX1	*FILE	*OBJOPR, *READ, *ADD, *UPDT, *DLT
capctllib <sup>1</sup>	IBMSNAP_REGISTERX2	*FILE	*OBJOPR, *READ, *ADD, *UPDT, *DLT
capctllib <sup>1</sup>	IBMSNAP_REG_EXT	*FILE	*OBJOPR, *READ, *ADD, *UPDT, *DLT
capctllib <sup>1</sup>	IBMSNAP_REG_EXTX	*FILE	*OBJOPR, *READ, *ADD, *UPDT, *DLT
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTL	*FILE	*OBJOPR, *READ
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTLX	*FILE	*OBJOPR, *READ
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTLX1	*FILE	*OBJOPR, *READ
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTLX2	*FILE	*OBJOPR, *READ
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTLX3	*FILE	*OBJOPR, *READ
ASN	ASN4B*	*SQLPKG	*USE
ASN	ASN4C*	*SQLPKG	*USE

**Remarque :**

1. L'entrée *capctllib* dans la colonne Bibliothèque fait référence à la valeur transmise au paramètre **CAPCTLLIB** de la commande **GRTDPRAUT** ; cette commande met à jour les droits d'accès à une seule bibliothèque de contrôle Capture à la fois.

La table suivante dresse la liste des droits d'accès accordés lorsque vous indiquez le paramètre AUT(\*SUBSCRIBER) dans la commande **GRTDPRAUT**.

Tableau 45. Droits d'accès accordés avec GRTDPRAUT AUT(\*SUBSCRIBER)

Bibliothèque	Objet	Type	Droits d'accès
QSYS	ASN	*LIB	*OBJOPR, *READ, *ADD, *EXECUTE
QSYS	capctllib	*LIB	*OBJOPR, *READ, *ADD, *EXECUTE
ASN	IBMSNAP_SUBS_SET	*FILE	*CHANGE
ASN	IBMSNAP_SUBS_COLS	*FILE	*CHANGE
ASN	IBMSNAP_SUBS_EVENT	*FILE	*CHANGE
ASN	IBMSNAP_SUBS_STMTS	*FILE	*CHANGE
ASN	IBMSNAP_SUBS_MEMBR	*FILE	*CHANGE

Tableau 45. Droits d'accès accordés avec **GRTDPRAUT AUT(\*SUBSCRIBER)** (suite)

Bibliothèque	Objet	Type	Droits d'accès
capctllib <sup>1</sup>	IBMSNAP_REGISTER	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REG_EXT	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTL	*FILE	*OBJOPR, *READ, *DLT, *ADD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTLX	*FILE	*USE
ASN	ASN4A*	*SQLPKG	*USE
ASN	ASN4U*	*SQLPKG	*USE

**Remarque :**

1. L'entrée *capctllib* dans la colonne Bibliothèque fait référence à la valeur transmise au paramètre **CAPCTLLIB** de la commande **GRTDPRAUT** ; cette commande met à jour les droits d'accès à une seule bibliothèque de contrôle Capture à la fois.

La table suivante dresse la liste des droits d'accès accordés lorsque vous indiquez le paramètre **AUT(\*CAPTURE)** dans la commande **GRTDPRAUT**.

Tableau 46. Droits d'accès accordés avec **GRTDPRAUT AUT(\*CAPTURE)**

Bibliothèque	Objet	Type	Droits d'accès
QSYS	capctllib	*LIB	*OBJOPR, *OBJMGT, *READ, *EXECUTE
QSYS	QDP4	*LIB	*OBJOPR, *ADD, *READ, *EXECUTE
capctllib <sup>1</sup>	QZSN	*MSGQ	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_REGISTER	*FILE	*OBJOPR, *OBJMGT, *READ, *ADD, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REGISTERX	*FILE	*OBJOPR, *OBJMGT, *READ, *ADD, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REGISTERX1	*FILE	*OBJOPR, *OBJMGT, *READ, *ADD, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REGISTERX2	*FILE	*OBJOPR, *OBJMGT, *READ, *ADD, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REG_EXT	*FILE	*OBJOPR, *OBJMGT, *READ, *ADD, *UPD, *EXECUTE

Tableau 46. Droits d'accès accordés avec *GRTDPRAUT AUT(\*CAPTURE)* (suite)

Bibliothèque	Objet	Type	Droits d'accès
capctllib <sup>1</sup>	IBMSNAP_REG_EXTX	*FILE	*OBJOPR, *OBJMGT, *READ, *ADD, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTL	*FILE	*OBJOPR, *OBJMGT, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTLX	*FILE	*OBJOPR, *OBJMGT, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTLX1	*FILE	*OBJOPR, *OBJMGT, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTLX2	*FILE	*OBJOPR, *OBJMGT, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTLX3	*FILE	*OBJOPR, *OBJMGT, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_CAPTRACE	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_CAPTRACEX	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_RESTART	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_RESTARTX	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_AUTHTKN	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_AUTHTKNX	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_UOW	*FILE	*OBJOPR, *OBJMGT, *READ, *UPD, *DLT, *ADD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_UOW_IDX	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_PRUNE_SET	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_PRUNE_SETX	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_CAPPARMS	*FILE	*READ, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_SIGNAL	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_SIGNALX	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_CAPMON	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_CAPMONX	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_PRUNE_LOCK	*FILE	*CHANGE
ASN	ASN4B*	*SQLPKG	*USE
ASN	ASN4C*	*SQLPKG	*USE
ASN	QZS8CTLBLK	*USRSPC	*CHANGE

**Remarque :**

1. L'entrée *capctllib* dans la colonne Bibliothèque fait référence à la valeur transmise au paramètre **CAPCTLLIB** de la commande **GRTDPRAUT** ; cette commande met à jour les droits d'accès à une seule bibliothèque de contrôle Capture à la fois.

La table suivante dresse la liste des droits d'accès accordés lorsque vous indiquez le paramètre AUT(\*APPLY) dans la commande **GRTDPRAUT**.

Tableau 47. Droits d'accès accordés avec GRTDPRAUT AUT(\*APPLY)

Bibliothèque	Objet	Type	Droits d'accès
QSYS	ASN	*LIB	*OBJOPR, *READ, *EXECUTE
QSYS	capctllib	*LIB	*OBJOPR, *READ, *EXECUTE
QDP4	QZSNAPV2	*PGM	*OBJOPR, *READ, *OBMGT, *OBJALTER, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REGISTER	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REGISTERX	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REGISTERX1	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REGISTERX2	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REGISTER_EXT	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_REGISTER_EXTX	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_SIGNAL	*FILE	*OBJOPR, *READ, *UPD, *ADD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_SIGNALX	*FILE	*OBJOPR, *READ, *UPD, *ADD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_PRUNE_LOCK	*FILE	*CHANGE
capctllib <sup>1</sup>	IBMSNAP_UOW	*FILE	*OBJOPR, *READ, *UPD, *ADD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_PRUNCNTL	*FILE	*OBJOPR, *READ, *UPD, *ADD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_AUTHTKN	*FILE	*OBJOPR, *READ, *UPD, *ADD, *EXECUTE
capctllib <sup>1</sup>	IBMSNAP_AUTHTKNX	*FILE	*OBJOPR, *READ, *UPD, *ADD, *EXECUTE
ASN	IBMSNAP_SUBS_SET	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
ASN	IBMSNAP_SUBS_SETX	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
ASN	IBMSNAP_APPLYTRAIL	*FILE	*OBJOPR, *READ, *UPD, *ADD, *EXECUTE



Tableau 47. Droits d'accès accordés avec `GRTDPRAUT AUT(*APPLY)` (suite)

Bibliothèque	Objet	Type	Droits d'accès
ASN	IBMSNAP_APPLYTRACE	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
ASN	IBMSNAP_APPLYTRACX	*FILE	*OBJOPR, *READ, *UPD, *EXECUTE
ASN	IBMSNAP_SUBS_COLS	*FILE	*USE
ASN	IBMSNAP_SUBS_EVENT	*FILE	*USE
ASN	IBMSNAP_SUBS_STMTS	*FILE	*USE
ASN	IBMSNAP_SUBS_MEMBR	*FILE	*USE
ASN	ASN4A*	*SQLPKG	*USE
ASN	ASN4U*	*SQLPKG	*USE
ASN	IBMSNAP_APPLY_JOB	*FILE	*OBJOPR, *READ, *UPD, *ADD, *EXECUTE

**Remarque :**

1. L'entrée *capctllib* dans la colonne Bibliothèque fait référence à la valeur transmise au paramètre **CAPCTLLIB** de la commande **GRTDPRAUT** ; cette commande met à jour les droits d'accès à une seule bibliothèque de contrôle Capture à la fois.

## Exemples pour GRTDPRAUT

Les exemples suivants illustrent la manière d'utiliser la commande **GRTDPRAUT**.

### Exemple 1 :

Pour autoriser un utilisateur nommé USER1 à définir et à modifier des enregistrements :

```
GRTDPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*REGISTRAR)
```

### Exemple 2 :

Pour autoriser un utilisateur nommé USER1 à définir et à modifier des ensembles d'abonnements :

```
GRTDPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*SUBSCRIBER)
```

### Exemple 3 :

Pour autoriser un utilisateur nommé USER1 à exécuter des programmes Capture :

```
GRTDPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*CAPTURE)
```

### Exemple 4 :

Pour autoriser un utilisateur nommé USER1 à définir et à modifier des ensembles d'abonnements existants associés au qualificatif Apply A1 :

```
GRTDPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*SUBSCRIBER) APYQUAL(A1)
```

### Exemple 5 :

Pour autoriser un utilisateur à exécuter le programme Apply sur le système du serveur de contrôle pour tous les ensembles d'abonnements associés au qualificatif Apply A1, lorsque le serveur cible est identique au serveur de contrôle :

1. Exécutez la commande suivante sur le système sur lequel s'exécutera le programme Apply :  
`GRTPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*APPLY) APYQUAL(A1)`
2. Exécutez la commande **GRTPRAUT** appropriée sur le système du serveur source :
  - Si le travail du serveur d'application sur le serveur source utilisé par le programme Apply s'exécute sous le profil utilisateur USER1, exécutez la commande suivante sur les systèmes du serveur source :  
`GRTPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*APPLY) APYQUAL(A1)`
  - Si le travail du serveur d'application sur le serveur source utilisé par le programme Apply s'exécute sous un profil utilisateur différent, par exemple, QUSER, la commande est la suivante :  
`GRTPRAUT CAPCTLLIB(ASN) USER(QUSER) AUT(*APPLY) APYQUAL(A1)`

## INZDPRCAP : réinitialisation de la capture DPR (System i)

Utilisez la commande Initialize DPR Capture (**INZDPRCAP**) pour initialiser le programme Capture en demandant au programme Capture de travailler avec une liste mise à jour des tables source.

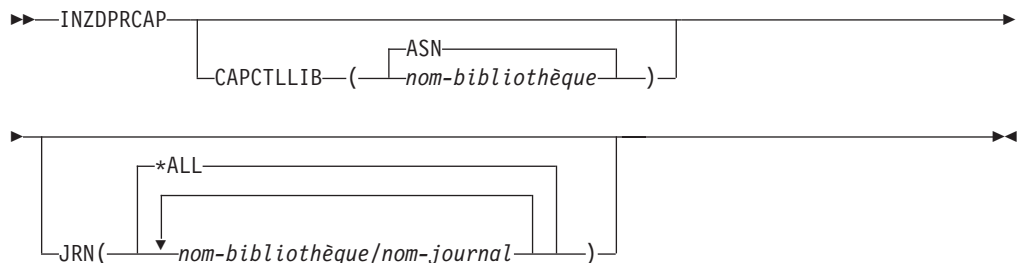
Les tables source sous le contrôle d'un programme Capture peuvent changer pendant que le programme Capture est en cours d'exécution. Utilisez la commande **INZDPRCAP** pour veiller à ce que le programme Capture traite les sources de réplication les plus à jour.

Pour émettre cette commande, le programme Capture doit être en cours d'exécution.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe



Le tableau 48, à la page 375 dresse la liste des paramètres d'appel.

Tableau 48. Définitions des paramètres de commande INZDPRCAP pour System i

Paramètre	Définition et invites
<b>CAPCTLLIB</b>	<p>Spécifie le schéma Capture, qui est le nom de la bibliothèque dans laquelle résident les tables de contrôle Capture.</p> <p><b>ASN(valeur par défaut)</b> Les tables de contrôle Capture se trouvent dans la bibliothèque ASN. La bibliothèque ASN est la bibliothèque par défaut.</p> <p><i>nom-bibliothèque</i> Nom d'une bibliothèque contenant les tables de contrôle de Capture.</p>
<b>JRN</b>	<p>Spécifie un sous-ensemble de 50 journaux maximum avec lequel vous voulez que le programme Capture travaille. Le programme Capture commence le traitement de toutes les tables source actuellement journalisées dans ce journal.</p> <p><b>*ALL (valeur par défaut)</b> Le programme Capture travaille avec tous les journaux.</p> <p><i>nom-bibliothèque/nom-journal</i> Nom qualifié du journal avec lequel vous souhaitez que le programme Capture travaille.</p>

## Exemples pour INZDPRCAP

Les exemples suivants illustrent la manière d'utiliser la commande **INZDPRCAP**.

### Exemple 1 :

Pour initialiser un programme Capture à l'aide du journal QSQJRN sous une bibliothèque intitulée TRAINING :

```
INZDPRCAP CAPCTLLIB(ASN) JRN(TRAINING/QSQJRN)
```

Les tables de contrôle Capture se trouvent dans le schéma ASN.

### Exemple 2 :

Pour initialiser un programme Capture qui travaille avec tous les journaux :

```
INZDPRCAP CAPCTLLIB(BSN) JRN(*ALL)
```

Les tables de contrôle Capture se trouvent dans un schéma appelé BSN.

---

## OVRDPRCAPA : substitution des attributs DPR Capture (System i)

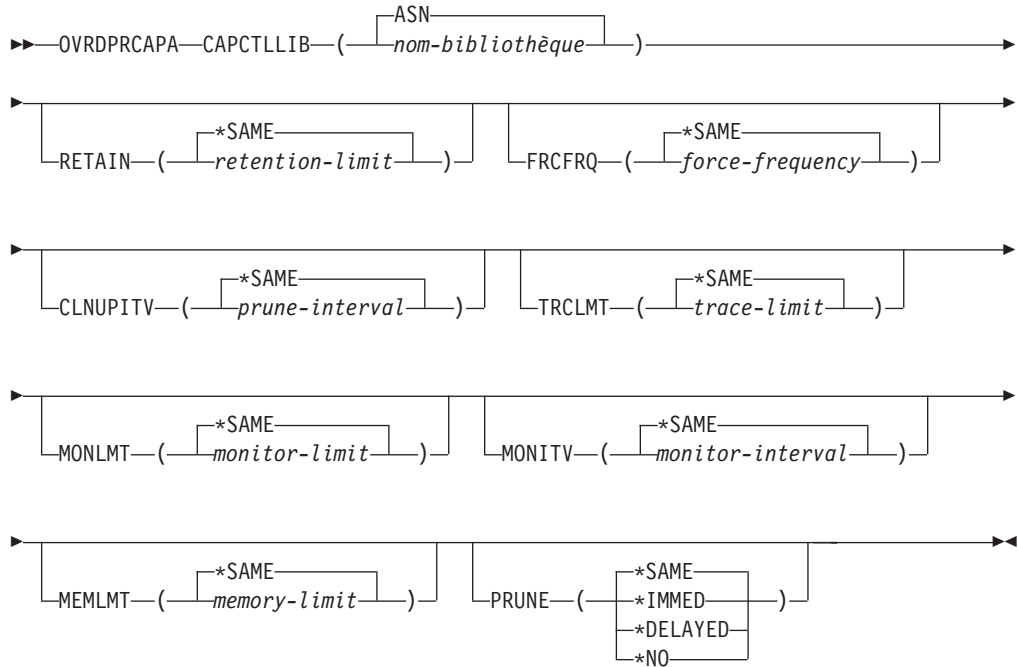
Utilisez la commande Override DPR Capture attributs (**OVRDPRCAPA**) pour modifier le comportement d'un programme Capture en cours d'exécution.

Cette commande modifie le comportement du programme en substituant les valeurs qui ont été transmises au programme Capture à partir de la table IBMSNAP\_CAPPARMS ou à partir de la commande **STRDPRCAP** lorsque le programme Capture a été lancé.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

## Syntaxe



Le tableau 49 dresse la liste des paramètres d'appel.

Tableau 49. Définitions des paramètres de commande OVRDPRCAPA pour System i

Paramètre	Définition et invites
<b>CAPCTLLIB</b>	<p>Spécifie le schéma Capture, qui est le nom de la bibliothèque dans laquelle résident les tables de contrôle Capture. Cette bibliothèque comprend la table IBMSNAP_REGISTER, qui stocke les informations d'enregistrement des tables source. Ce paramètre est obligatoire.</p> <p><b>ASN (valeur par défaut)</b>            Les tables de contrôle Capture se trouvent dans la bibliothèque ASN.</p> <p><i>nom-bibliothèque</i>            Nom d'une bibliothèque contenant les tables de contrôle de Capture. Vous pouvez créer cette bibliothèque à l'aide de la commande <b>CRTDPRTBL</b>, avec le paramètre <b>CAPCTLLIB</b>.</p>

Tableau 49. Définitions des paramètres de commande OVRDPRCAPA pour System i (suite)

Paramètre	Définition et invites
<b>RETAIN</b>	<p>Indique le nombre de minutes pendant lesquelles les données sont conservées dans les tables de données de modification (CD), d'unités d'oeuvre (UOW), IBMSNAP_SIGNAL, et IBMSNAP_AUTHTKN avant que les données ne soient supprimées.</p> <p>Cette valeur fonctionne avec la valeur de paramètre CLNUPITV à partir de la commande Start DPR Capture (<b>STRDPRCAP</b>). Avant tout, le programme Capture supprime toute ligne de table CD, UOW, IBMSNAP_SIGNAL, ou IBMSNAP_AUTHTKN qui est antérieure au programme Apply le plus ancien actuellement en cours d'exécution. Puis, une nouvelle ligne ou une ligne restante de la table CD, UOW, IBMSNAP_SIGNAL, ou IBMSNAP_AUTHTKN est ensuite supprimée lorsque son âge atteint la valeur du paramètre <b>RETAIN</b>.</p> <p>Veillez à ce que les intervalles Apply soient définis afin de copier les informations modifiées avant que les données n'atteignent cette valeur de paramètre <b>RETAIN</b>, et ce, afin d'éviter des données incohérentes dans vos tables. Si les données deviennent incohérentes, le programme Apply effectue une régénération intégrale.</p> <p>La valeur par défaut est de 10 080 minutes (sept jours). Le maximum est de 35000000 minutes.</p> <p><b>*SAME (valeur par défaut)</b>            Cette valeur n'est pas modifiée.</p> <p><i>retention-limit</i>            La nouvelle valeur de durée de conservation.</p>
<b>FRCFRQ</b>	<p>Indique la fréquence (de 30 à 600 secondes) selon laquelle le programme Capture écrit les modifications dans les tables de données de modification (CD) et d'unités d'oeuvre (UOW).</p> <p>Le programme Capture met ces modifications à la disposition du programme Apply soit lorsque les mémoires tampons sont remplies, soit lorsque la limite de temps <b>FRCFRQ</b> expire, quel que soit le plus tôt. Cette valeur de paramètre affecte la durée nécessaire au programme Capture pour réagir aux modifications à partir de la commande Initialize DPR Capture (<b>INZDPRCAP</b>).</p> <p>Utilisez ce paramètre pour rendre les modifications immédiatement disponibles pour le programme Apply sur les serveurs avec un faible taux de modifications de table source. La valeur de paramètre <b>FRCFRQ</b> est une valeur globale utilisée pour toutes les tables sources enregistrées. La configuration de la valeur <b>FRCFRQ</b> sur un faible nombre peut affecter les performances du système.</p> <p>La valeur par défaut est de 30 secondes.</p> <p><b>*SAME (valeur par défaut)</b>            Cette valeur n'est pas modifiée.</p> <p><i>force-frequency</i>            le nouveau nombre de secondes pendant lesquelles le programme Capture conserve les modifications de la table de données de modification et des unités d'oeuvre dans l'espace de la mémoire tampon avant de mettre ces modifications à la disposition du programme Apply.</p>

Tableau 49. Définitions des paramètres de commande OVRDPRCAPA pour System i (suite)

Paramètre	Définition et invites
<b>CLNUPITV</b>	<p>Indique la durée maximale (en heures) avant que le programme Capture n'élague les anciens enregistrements des tables de modification de données (CD), des tables d'unités d'oeuvre (UOW), et des tables IBMSNAP_SIGNAL, IBMSNAP_CAPMON, IBMSNAP_CAPTRACE, et IBMSNAP_AUTHTKN.</p> <p>Ce paramètre fonctionne avec le paramètre <b>RETAIN</b> pour contrôler l'élagage des tables CD, UOW, IBMSNAP_SIGNAL, et IBMSNAP_AUTHTKN, avec le paramètre <b>MONLMT</b> pour contrôler l'élagage de la table IBMSNAP_CAPMON, et avec le paramètre <b>TRCLMT</b> pour contrôler l'élagage de la table IBMSNAP_CAPTRACE.</p> <p>(Utilisez la commande <b>STRDPRCAP</b> pour définir les paramètres <b>RETAIN</b>, <b>MONLMT</b>, et <b>TRCLMT</b> pour un programme Capture.</p> <p>La valeur du paramètre <b>CLNUPITV</b> est automatiquement convertie d'heures en secondes et est stockée dans la colonne PRUNE_INTERVAL de la table IBMSNAP_CAPPARMS.</p> <p><b>*SAME (valeur par défaut)</b>            Cette valeur d'attribut Capture n'est pas modifiée.</p> <p><i>prune-interval</i>            L'intervalle d'élagage exprimé en nombre d'heures spécifique (1 à 100).</p>
<b>TRCLMT</b>	<p>Indique la limite de trace, laquelle indique la fréquence à laquelle la table IBMSNAP_CAPTRACE est élaguée.</p> <p><b>*SAME (valeur par défaut)</b>            Le programme Capture continue d'utiliser la valeur de limite de trace actuelle.</p> <p><i>trace-limit</i>            Le nombre de minutes entre chaque opération d'élagage de la table IBMSNAP_CAPTRACE.</p>
<b>MONLMT</b>	<p>Indique la limite de contrôle, laquelle indique la fréquence à laquelle la table IBMSNAP_CAPMON est élaguée.</p> <p><b>*SAME (valeur par défaut)</b>            Le programme Capture continue d'utiliser le nombre maximal de contrôles actuel.</p> <p><i>monitor-limit</i>            Le nombre de minutes entre chaque opération d'élagage de la table IBMSNAP_CAPMON.</p>
<b>MONITV</b>	<p>Indique l'intervalle de contrôle (en secondes), lequel indique la fréquence à laquelle le programme Capture insère des lignes dans la table IBMSNAP_CAPMON.</p> <p><b>*SAME (valeur par défaut)</b>            Le programme Capture continue d'utiliser l'intervalle de contrôle actuel.</p> <p><i>monitor-interval</i>            Le nombre de secondes entre l'insertion de ligne dans la table IBMSNAP_CAPMON. L'intervalle de contrôle doit être d'au moins 120 secondes (deux minutes). Si vous saisissez un nombre inférieur à 120, cette commande définit automatiquement cette valeur de paramètre sur 120.</p>

Tableau 49. Définitions des paramètres de commande OVRDPRCAPA pour System i (suite)

Paramètre	Définition et invites
MEMLMT	<p>Spécifie la taille maximale (en mégaoctets) de mémoire que le travail du journal Capture peut utiliser.</p> <p><b>*SAME (valeur par défaut)</b> Le programme Capture continue d'utiliser la valeur de limite de mémoire actuelle.</p> <p><i>memory-limit</i> Le nombre maximal de mégaoctets pour la mémoire.</p>
PRUNE	<p>Utilisez ce paramètre pour modifier la façon dont le programme Capture élague les lignes des tables de données de modification (CD), des unités d'oeuvre (UOW), IBMSNAP_SIGNAL, IBMSNAP_CAPMON, IBMSNAP_CAPTRACE, et IBMSNAP_AUTHTKN.</p> <p><b>*SAME (valeur par défaut)</b> Le programme Capture continue d'utiliser les paramètres d'élagage que vous avez spécifiés lorsque vous avez lancé la commande STRDPRCAP.</p> <p><b>*IMMED</b> Le programme Capture commence immédiatement l'élagage des tables, indépendamment de la valeur du paramètre CLNUPITV que vous avez spécifiée lorsque vous avez lancé la commande STRDPRCAP.</p> <p><b>*DELAYED</b> Le programme Capture élague les anciennes lignes à la fin de l'intervalle d'élagage spécifié.</p> <p>PRUNE(*DELAYED) n'affecte pas la fréquence d'élagage si vous définissez la deuxième partie du paramètre CLNUPITV sur *IMMED ou *DELAYED dans la commande STRDPRCAP. Toutefois, PRUNE(*DELAYED) lance effectivement l'élagage si vous définissez la deuxième partie du paramètre CLNUPITV sur *NO lorsque vous avez lancé la commande STRDPRCAP.</p> <p><b>*NO</b> Le programme Capture ne lance pas l'élagage. Cette valeur remplace la configuration du paramètre CLNUPITV à partir de la commande STRDPRCAP.</p>

## Exemples pour OVRDPRCAPA

Les exemples suivants illustrent comment utiliser la commande OVRDPRCAPA.

### Exemple 1 :

Pour modifier les paramètres d'élagage des tables CD, UOW, IBMSNAP\_SIGNAL, IBMSNAP\_CAPMON, IBMSNAP\_CAPTRACE, et IBMSNAP\_AUTHTKN (qui résident dans la bibliothèque ASN par défaut) et pour modifier l'intervalle de contrôle IBMSNAP\_CAPMON et la limite de mémoire des travaux de journal Capture dans un programme Capture en cours d'exécution :

```
OVRDPRCAPA CAPCTLLIB(ASN) CLNUPITV(12) MONITV(600) MEMLMT(64)
```

### Exemple 2 :

Pour lancer l'élagage des tables CD, UOW, IBMSNAP\_SIGNAL, IBMSNAP\_CAPMON, IBMSNAP\_CAPTRACE, et IBMSNAP\_AUTHTKN qui résident dans la bibliothèque BSN :

```
OVRDPRCAPA CAPCTLLIB(BSN) PRUNE(*IMMED)
```

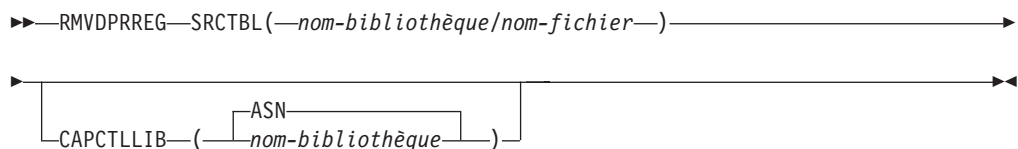
## RMVDPRREG : suppression d'un enregistrement DPR (System i)

Utilisez la commande Remove DPR registration (**RMVDPRREG**) pour supprimer une table source unique de la table IBMSNAP\_REGISTER afin que la table source ne soit plus utilisée pour la réplication.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe



Le tableau 50 dresse la liste des paramètres d'appel.

Tableau 50. Définitions des paramètres de commande RMVDPRREG pour System i

Paramètre	Définition et invites
<b>SRCTBL</b>	Identifie l'enregistrement que vous voulez supprimer. Il s'agit d'un paramètre obligatoire.  <i>nom-bibliothèque/nom-fichier</i> Nom qualifié du fichier enregistré.
<b>CAPCTLLIB</b>	Spécifie le schéma Capture, qui est le nom de la bibliothèque dans laquelle résident les tables de contrôle Capture.  <b>ASN (valeur par défaut)</b> Les tables de contrôle Capture se trouvent dans la bibliothèque ASN.  <i>nom-bibliothèque</i> Indique le nom d'un schéma contenant les tables de contrôle de Capture.

### Exemples pour RMVDPRREG

Les exemples suivants illustrent comment utiliser la commande **RMVDPRREG**.

#### Exemple 1 :

Pour supprimer l'enregistrement de la table source nommée EMPLOYEE de la bibliothèque RH dans le schéma ASN Capture par défaut :

```
RMVDPRREG SRCTBL(HR/EMPLOYEE)
```



## Exemple 2 :

Pour supprimer l'enregistrement de la table source nommée SALES de la bibliothèque DEPT sous un schéma Capture nommé BSN :

```
RMVDPRREG SRCTBL(DEPT/SALES) CAPCTLLIB(BSN)
```

---

## RMVDPRSUB : suppression d'un ensemble d'abonnements DPR (System i)

Utilisez la commande Remove DPR subscription set (**RMVDPRSUB**) pour supprimer un ensemble d'abonnements. Si vous définissez le paramètre **RMVMBRS** sur \*YES, cette commande supprime l'ensemble d'abonnements ainsi que tous ses membres.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe

```
▶▶ RMVDPRSUB—APYQUAL—(—apply-qualifier—)—SETNAME—(—nom-ensemble—)—▶▶  
  
▶ [CTLSVR—(—[*LOCAL]—)] [nom-bdr] [RMVREG—(—[*NO]—) [*YES]—]▶▶  
  
▶ [DLTTGTTBL—(—[*NO]—) [*YES]—] [RMVMBRS—(—[*NO]—) [*YES]—]▶▶
```

Le tableau 51 dresse la liste des paramètres d'appel.

Tableau 51. Définitions des paramètres de commande RMVDPRSUB pour System i

Paramètre	Définition et invites
APYQUAL	Spécifie le qualificatif Apply utilisé par le programme Apply pour identifier l'ensemble d'abonnements. Ce paramètre est obligatoire. <i>apply-qualifier</i> Nom du qualificatif Apply.
SETNAME	Spécifie le nom de l'ensemble d'abonnements. Ce paramètre est obligatoire. <i>nom-ensemble</i> Nom de l'ensemble d'abonnements. Vous recevez un message d'erreur si vous saisissez un nom d'ensemble d'abonnements qui n'existe pas pour le qualificatif Apply spécifié.

Tableau 51. Définitions des paramètres de commande *RMVDPRSUB* pour System i (suite)

Paramètre	Définition et invites
<b>CTLSVR</b>	<p>Spécifie le nom de la base de données relationnelle du système contenant les tables de contrôle Apply.</p> <p><b>*LOCAL (par défaut)</b> Les tables de contrôle Apply résident localement (sur l'ordinateur sur lequel vous exécutez la commande <b>RMVDPRSUB</b>).</p> <p><i>nom-bdr</i> Le nom de la base de données relationnelle où résident les tables de contrôle Apply. Vous pouvez utiliser la commande Work with RDB Directory Entries (<b>WRKRBDIRE</b>) pour trouver ce nom.</p>
<b>RMVREG</b>	<p>Indique si cette commande supprime les enregistrements associés aux tables cible de tous les membres de l'ensemble d'abonnements dans l'ensemble d'abonnements. N'utilisez ce paramètre que si vous avez défini le paramètre <b>RMVMBRS</b> sur *YES.</p> <p><b>*NO (valeur par défaut)</b> Les enregistrements ne sont pas supprimés.</p> <p><b>*YES</b> Les enregistrements sont supprimés.</p>
<b>DLTTGTTBL</b>	<p>Indique si cette commande supprime les tables cible des membres de l'ensemble d'abonnements une fois que celui-ci est supprimé. N'utilisez ce paramètre que si vous avez défini le paramètre <b>RMVMBRS</b> sur *YES.</p> <p><b>*NO (valeur par défaut)</b> Les tables cible ne sont pas supprimées.</p> <p><b>*YES</b> Les tables cible sont supprimées.</p>
<b>RMVMBRS</b>	<p>Indique si cette commande supprime l'ensemble d'abonnements ainsi que tous les membres de cet ensemble d'abonnements.</p> <p><b>*NO (valeur par défaut)</b> L'ensemble d'abonnements n'est pas supprimé s'il y a des membres existant dans l'ensemble d'abonnements.</p> <p><b>*YES</b> L'ensemble d'abonnements ainsi que tous les membres de cet ensemble d'abonnements sont supprimés.</p>

## Exemples pour **RMVDPRSUB**

Les exemples suivants illustrent comment utiliser la commande **RMVDPRSUB**.

### Exemple 1 :

Pour supprimer un ensemble d'abonnements nommé **SETHR** et qui ne contient aucun membre d'ensemble d'abonnements :

```
RMVDPRSUB APYQUAL(AQHR) SETNAME(SETHR)
```

### Exemple 2 :

Pour supprimer un ensemble d'abonnements **SETHR** ainsi que tous les membres de l'ensemble d'abonnements :

```
RMVDPRSUB APYQUAL(AQHR) SETNAME(SETHR) RMVMBRS(*YES)
```

### Exemple 3 :

Pour supprimer un ensemble d'abonnements nommé SETHR, ainsi que tous les membres de cet ensemble d'abonnements, et les enregistrements associés :

```
RMVDPRSUB APYQUAL(AQHR) SETNAME(SETHR) RMVREG(*YES) RMVMBRS(*YES)
```

---

## RMVDPRSUBM : suppression d'un membre d'un ensemble d'abonnements DPR (System i)

Utilisez la commande Remove DPR subscription-set member (**RMVDPRSUBM**) pour supprimer un seul membre d'un ensemble d'abonnements d'un ensemble d'abonnements.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe

```
▶▶—RMVDPRSUBM—APYQUAL—(—apply-qualifier—)—SETNAME—(—nom-ensemble—)—▶▶
▶—TGTTBL—(—nom-bibliothèque/nom-fichier—)—▶▶
└─CTLSVR—(—*LOCAL—)
└─nom-bdr—)
▶▶
└─RMVREG—(—*NO—)
└─*YES—)
└─DLTTGTTBL—(—*NO—)
└─*YES—)▶▶
```

Le tableau 52 dresse la liste des paramètres d'appel.

Tableau 52. Définitions des paramètres de commande RMVDPRSUBM pour System i

Paramètre	Définition et invites
<b>APYQUAL</b>	Spécifie le qualificatif Apply utilisé par le programme Apply pour identifier l'ensemble d'abonnements. Ce paramètre est obligatoire. <i>apply-qualifier</i> Nom du qualificatif Apply.
<b>SETNAME</b>	Spécifie le nom de l'ensemble d'abonnements. Ce paramètre est obligatoire. <i>nom-ensemble</i> Nom de l'ensemble d'abonnements. Vous recevez un message d'erreur si vous saisissez un nom d'ensemble d'abonnements qui n'existe pas pour le qualificatif Apply spécifié.
<b>TGTTBL</b>	Spécifie la table cible enregistrée pour le membre de l'ensemble d'abonnements. Ce paramètre est obligatoire. <i>nom-bibliothèque/nom-fichier</i> Le nom qualifié de la table cible.

Tableau 52. Définitions des paramètres de commande RMVDPRSUBM pour System i (suite)

Paramètre	Définition et invites
CTLSVR	<p>Spécifie le nom de la base de données relationnelle du système contenant les tables de contrôle Apply.</p> <p><b>*LOCAL (par défaut)</b> Les tables de contrôle Apply résident localement (sur l'ordinateur sur lequel vous exécutez la commande <b>RMVDPRSUBM</b>).</p> <p><i>nom-bdr</i> Le nom de la base de données relationnelle où résident les tables de contrôle Apply. Vous pouvez utiliser la commande Work with RDB Directory Entries (<b>WRKRDBDIRE</b>) pour trouver ce nom.</p>
RMVREG	<p>Indique si cette commande supprime l'enregistrement qui est associé à la table cible pour le membre de l'ensemble d'abonnements.</p> <p><b>*NO (valeur par défaut)</b> L'enregistrement n'est pas supprimé.</p> <p><b>*YES</b> L'enregistrement est supprimé.</p>
DLTTGTTBL	<p>Indique si cette commande supprime la table cible du membre de l'ensemble d'abonnements une fois que celui-ci est supprimé.</p> <p><b>*NO (valeur par défaut)</b> La table cible n'est pas supprimée.</p> <p><b>*YES</b> La table cible est supprimée.</p>

## Exemples pour RMVDPRSUBM

Les exemples suivants illustrent comment utiliser la commande **RMVDPRSUBM**.

### Exemple 1 :

Pour supprimer un membre d'un ensemble d'abonnements, qui utilise une table cible nommée EMP, d'un ensemble d'abonnements SETEMP sur la base de données relationnelle nommée RMTRDB1 :

```
RMVDPRSUBM APYQUAL(AQHR) SETNAME(SETEMP) TGTTBL(TGTEMP/EMP) CTLSVR(RMTRDB1)
```

### Exemple 2 :

Pour supprimer un membre d'un ensemble d'abonnements de l'ensemble d'abonnements SETHR, supprimez l'enregistrement, puis supprimez la table :

```
RMVDPRSUBM APYQUAL(AQHR) SETNAME(SETHR) TGTTBL(TGTHR/YTD TAX) RMVREG(*YES)
DLTTGTTBL(*YES)
```

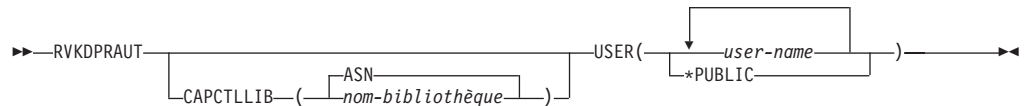
## RVKDPRAUT : révocation des droits d'accès(System i)

La commande Revoke DPR Authority (**RVKDPRAUT**) révoque les droits d'accès aux tables de contrôle de réplication de sorte que les utilisateurs ne puissent plus définir ou modifier les sources de réplication et les ensembles d'abonnements.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

## Syntaxe



Le tableau 53 dresse la liste des paramètres d'appel.

Tableau 53. Définitions des paramètres de commande RVKDPRAUT pour System i

Paramètre	Définition et invites
<b>CAPCTLLIB</b>	<p>Spécifie le schéma Capture, qui est le nom de la bibliothèque dans laquelle les droits d'accès de l'utilisateur ont été révoqués.</p> <p><b>ASN(valeur par défaut)</b> Les tables de contrôle Capture se trouvent dans la bibliothèque ASN.</p> <p><i>nom-bibliothèque</i> Nom de la bibliothèque contenant les tables de contrôle de replication.</p>
<b>USER</b>	<p>Indique les utilisateurs dont les droits d'accès sont révoqués. Ce paramètre est obligatoire.</p> <p><i>user-name</i> Indique les noms de 50 utilisateur au maximum dont les droits d'accès sont révoqués.</p> <p><b>*PUBLIC</b> Indique que les droits d'accès sont révoqués pour tous les utilisateurs sans droit d'accès spécifique, qui ne sont pas sur la liste d'autorisation, et dont le profil de groupe n'a aucun droit d'accès.</p>

## Notes sur l'utilisation

La commande renvoie un message d'erreur si l'une des conditions suivantes existe :

- Un utilisateur spécifié n'existe pas.
- L'utilisateur exécutant la commande n'est pas autorisé sur les profils utilisateur spécifiés.
- L'utilisateur exécutant la commande ne bénéficie pas de l'autorisation de révoquer les droits d'accès aux tables de contrôle DB2 DataPropagator pour System i.
- Les tables de contrôle DB2 DataPropagator pour System i n'existent pas.
- Les programmes Capture ou Apply sont en cours d'exécution.

## Exemples pour RVKDPRAUT

Les exemples suivants illustrent comment utiliser la commande RVKDPRAUT.

### Exemple 1 :

Pour révoquer le droit d'accès d'un utilisateur nommé HJONES sur les tables de contrôle dans la bibliothèque ASN :

```
RVKDPRAUT CAPCTLLIB(ASN) USER(HJONES)
```

**Exemple 2 :**

Pour révoquer les droits d'accès de tous les utilisateurs qui n'ont pas été spécifiés dans la commande **GRTDPRAUT** afin qu'ils ne puissent accéder aux tables de contrôle de la bibliothèque ASN :

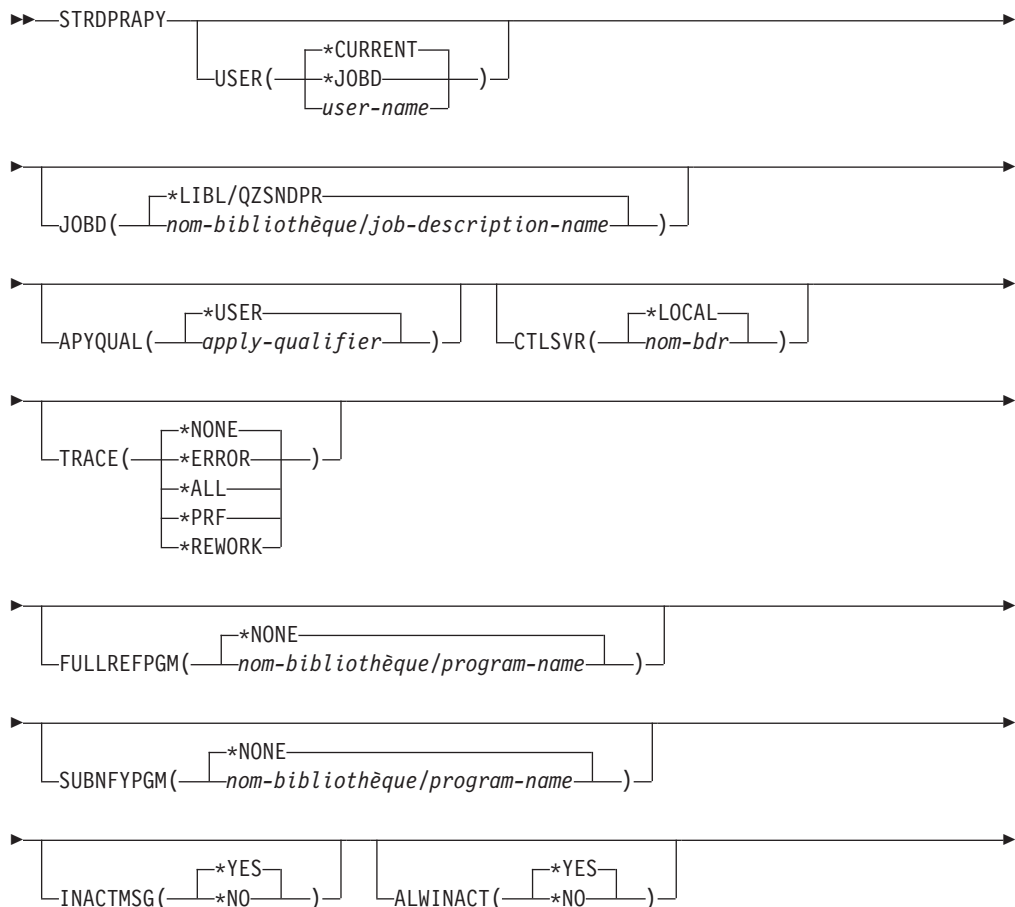
```
RVKDPRAUT CAPCTLLIB(ASN) USER(*PUBLIC)
```

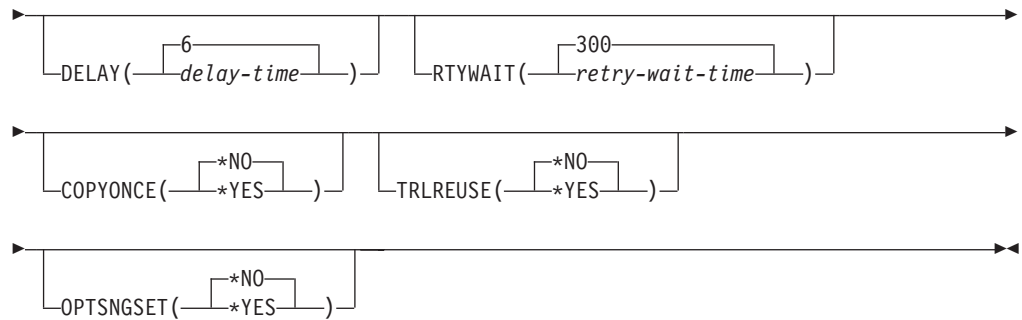
## STRDPRAPY : démarrage de Apply (System i)

Utilisez ma commande Start DPR Apply (**STRDPRAPY**) pour démarrer un programme Apply sur votre système local. Le programme Apply continue de s'exécuter jusqu'à ce que vous l'arrêtiez ou jusqu'à ce qu'il détecte une erreur irréparable.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.





Le tableau 54 dresse la liste des paramètres d'appel.

Tableau 54. Définitions des paramètres de commande STRDPRAPY pour System i

Paramètre	Définition et invites
<b>USER</b>	<p>Indique le nom de l'ID utilisateur pour lequel démarrer le programme Apply. Lorsque vous exécutez cette commande, vous devez être autorisé (avoir les droits d'accès *USE) au profil utilisateur spécifié ; le programme Apply s'exécute sous ce profil utilisateur spécifié.</p> <p>Les tables de contrôle sont situées dans la base de données relationnelle spécifiée par le paramètre <b>CTLSVR</b>. Les mêmes tables de contrôle sont utilisées indépendamment de la valeur spécifiée dans le paramètre <b>USER</b>.</p> <p><b>*CURRENT (valeur par défaut)</b> L'ID utilisateur associé au travail en cours est le même ID utilisateur associé à ce programme Apply.</p> <p><b>*JOBID</b> L'ID utilisateur spécifié dans la description du travail associé à ce programme Apply. La description de travail ne peut pas spécifier USER(*RQD).</p> <p><i>user-name</i> L'ID utilisateur associé à ce programme Apply. Les objets suivants fournis par IBM ne sont <i>pas</i> valides sur ce paramètre : QDBSHR, QDFTOWN, QDOC, QLPAUTO, QLPINSTALL, QRJE, QSECOFR, QSPL, QSYS, ou QTSTRQS.</p> <p>Lors de l'invite sur la commande <b>STRDPRAPY</b>, vous pouvez appuyer sur la touche F4 pour afficher une liste des utilisateurs qui ont défini des ensembles d'abonnements.</p>
<b>JOBID</b>	<p>Spécifie le nom de la description de travail à utiliser lorsque l'on soumet le programme Apply.</p> <p><b>*LIBL/QZSNDPR (valeur par défaut)</b> La description de travail par défaut fournie avec DB2 DataPropagator pour System i.</p> <p><i>nom-bibliothèque/job-description-name</i> Nom de la description de travail utilisée pour le programme Apply.</p>

Tableau 54. Définitions des paramètres de commande STRDPRAPY pour System i (suite)

Paramètre	Définition et invites
APYQUAL	<p>Indique le qualificatif Apply devant être utilisé par le programme Apply. Tous les ensembles d'abonnements qui sont regroupés avec ce qualificatif Apply sont exécutés par le programme Apply.</p> <p><b>*USER (valeur par défaut)</b> La valeur de paramètre <b>USER</b> que vous avez saisie est utilisée comme nom du qualificatif Apply.</p> <p><i>apply-qualifier</i> Le nom utilisé pour regrouper les ensembles d'abonnements qui doivent être exécutés par ce programme Apply. Pour le nom du qualificatif Apply, vous pouvez spécifier un maximum de 18 caractères. Ce nom suit les mêmes conventions de dénomination qu'un nom de base de données relationnelle.</p> <p>Lors de l'invite sur la commande <b>STRDPRAPY</b>, vous pouvez appuyer sur la touche F4 pour afficher une liste des noms de qualificatifs Apply avec des ensembles d'abonnements existants.</p>
CTLSVR	<p>Spécifie le nom de la base de données relationnelle du système contenant les tables de contrôle Apply.</p> <p><b>*LOCAL (par défaut)</b> Les tables de contrôle Apply résident localement (sur l'ordinateur sur lequel vous exécutez la commande <b>STRDPRAPY</b>).</p> <p><i>nom-bdr</i> Le nom de la base de données relationnelle où résident les tables de contrôle Apply. Vous pouvez utiliser la commande Work with RDB Directory Entries (<b>WRKRDBDIRE</b>) pour trouver ce nom.</p> <p>Lors de l'invite sur la commande <b>STRDPRAPY</b>, vous pouvez appuyer sur la touche F4 pour afficher une liste des noms RDB disponibles.</p>
TRACE	<p>Indique si le programme Apply doit générer une trace. Le programme Apply écrit les données de trace dans un fichier spoule appelé QPZSNATRC.</p> <p><b>*NONE (valeur par défaut)</b> Aucune trace n'est générée.</p> <p><b>*ERROR</b> La trace contient uniquement des informations sur les erreurs.</p> <p><b>*ALL</b> La trace contient les informations sur les erreurs et les flux d'exécution.</p> <p><b>*PRF</b> La trace contient des informations qui peuvent être utilisées pour analyser les performances à des phases différentes de l'exécution du programme Apply.</p> <p><b>*REWORK</b> La trace contient des informations sur des lignes qui ont été transformées par le programme Apply.</p>



Tableau 54. Définitions des paramètres de commande STRDPRAPY pour System i (suite)

Paramètre	Définition et invites
<b>FULLREFPGM</b>	<p>Indique si le programme Apply doit appeler une routine d'exit pour initialiser une table cible. Lorsque le programme Apply est prêt à effectuer une régénération intégrale d'une table cible, le programme Apply appelle la routine d'exit spécifiée au lieu d'exécuter la régénération intégrale en elle-même.</p> <p>Lorsqu'une routine d'exit de régénération intégrale est utilisée par le programme Apply, la valeur de la colonne ASNLOAD dans la table IBMSNAP_APPLYTRAIL est Y.</p> <p><b>*NONE (valeur par défaut)</b> Une routine d'exit de régénération intégrale n'est pas utilisée.</p> <p><i>nom-bibliothèque/program-name</i> Le nom qualifié du programme qui est appelé par le programme Apply exécutant une régénération intégrale d'une table cible. Par exemple, pour appeler le programme ASNLOAD dans la bibliothèque DATAPROP, le nom qualifié est DATAPROP/ASNLOAD.</p>
<b>SUBNFYPGM</b>	<p>Indique si le programme Apply doit appeler une routine d'exit lorsque le programme achève le traitement d'un ensemble d'abonnements. La saisie dans la routine d'exit comprend le nom de l'ensemble d'abonnements, le qualificatif Apply, l'état d'avancement ainsi que les statistiques avec le nombre de rejets.</p> <p>Le programme de notification vous permet d'examiner la table des unités d'oeuvre (UOW) afin de déterminer le moment auquel les transactions ont été rejetées et quand effectuer de nouvelles actions, telles que l'émission d'un message ou la génération d'un événement.</p> <p><b>*NONE (valeur par défaut)</b> Une routine d'exit n'est pas utilisée.</p> <p><i>nom-bibliothèque/program-name</i> Le nom qualifié du programme de routine d'exit appelé par le programme Apply lors du traitement d'un ensemble d'abonnements. Par exemple, pour appeler un programme APPLYDONE dans une bibliothèque DATAPROP, le nom qualifié est DATAPROP/APPLYDONE.</p>
<b>INACTMSG</b>	<p>Indique si le programme Apply doit générer un message chaque fois qu'il achève son travail et devient inactif pendant une certaine période.</p> <p><b>*YES (valeur par défaut)</b> Le programme Apply génère le message ASN1044 avant de commencer une période d'inactivité. Le message ASN1044 indique la durée pendant laquelle le programme Apply reste inactif.</p> <p><b>*NO</b> Aucun message n'est généré.</p>

Tableau 54. Définitions des paramètres de commande STRDPRAPY pour System i (suite)

Paramètre	Définition et invites
<b>ALWINACT</b>	<p>Indique si le programme Apply peut s'exécuter dans un état inactif (veille).</p> <p><b>*YES (valeur par défaut)</b> Le programme Apply est en veille s'il n'y a rien à traiter.</p> <p><b>*NO</b> Si le programme Apply n'a rien à traiter, le travail qui a soumis et lancé le programme Apply s'arrête.</p>
<b>DELAY</b>	<p>Indique le temps de retard (en secondes) à la fin de chaque cycle du programme Apply lorsque la réplication continue est utilisée.</p> <p><b>6 (valeur par défaut)</b> Le temps de retard est de six secondes.</p> <p><i>delay-time</i> Le temps de retard, saisi en tant que nombre entre 0 et 6 inclus.</p>
<b>RTYWAIT</b>	<p>Indique la durée (en secondes) pendant laquelle le programme Apply doit attendre après avoir rencontré une erreur avant de retenter l'opération qui a échoué.</p> <p><b>300 (valeur par défaut)</b> Le temps d'attente avant la nouvelle tentative est de 300 secondes (cinq minutes).</p> <p><i>retry-wait-time</i> Le temps d'attente, saisi en tant que nombre compris entre 0 et 35000000 inclus, avant que le programme Apply ne retente l'opération qui a échoué.</p>
<b>COPYONCE</b>	<p>Indique si le programme Apply exécute un cycle de copie pour chaque ensemble d'abonnements éligible au moment où le programme Apply est appelé. Ensuite, le programme Apply s'arrête. Un ensemble d'abonnements éligible remplit les critères suivants :</p> <ul style="list-style-type: none"> <li>• (ACTIVATE &gt; 0) dans la table IBMSNAP_SUBS_SET. Lorsque la valeur de la colonne ACTIVATE est supérieure à zéro, l'ensemble d'abonnements est actif de manière permanente ou est utilisé pour un processus d'abonnement unique.</li> <li>• (REFRESH_TYPE = R ou B) ou (REFRESH_TYPE = E et l'événement spécifié s'est produit). La valeur de la colonne REFRESH_TYPE est stockée dans la table IBMSNAP_SUBS_SET.</li> </ul> <p>La limite MAX_SYNCH_MINUTES de la table IBMSNAP_SUBS_SET et l'horodatage END_OF_PERIOD de la table IBMSNAP_SUBS_EVENT sont pris en compte s'ils sont spécifiés.</p> <p><b>*NO (valeur par défaut)</b> Le programme Apply n'exécute pas un cycle de copie pour chaque ensemble d'abonnements éligible.</p> <p><b>*YES</b> Le programme Apply exécute un cycle de copie pour chaque ensemble d'abonnements admissible et se termine.</p>

Tableau 54. Définitions des paramètres de commande STRDPRAPY pour System i (suite)

Paramètre	Définition et invites
TRLREUSE	<p>Indique si le programme Apply vide la table IBMSNAP_APPLYTRAIL lorsqu'il démarre.</p> <p><b>*NO (valeur par défaut)</b> Le programme Apply ne vide pas la table IBMSNAP_APPLYTRAIL lorsqu'il démarre.</p> <p><b>*YES</b> Le programme Apply vide la table IBMSNAP_APPLYTRAIL lorsqu'il démarre.</p>
OPTSNGSET	<p>Indique si les performances du programme Apply sont optimisées si un seul ensemble d'abonnements est traité. Ce paramètre n'appartient pas aux tables cible de réplication.</p> <p>Si vous avez défini ce paramètre sur *YES, le programme Apply extrait les membres et les colonnes d'un ensemble d'abonnements une seule fois et réutilise ces informations extraites lors du traitement de ce même ensemble d'abonnements dans deux cycles de traitement consécutifs ou plus.</p> <p><b>*NO (valeur par défaut)</b> Les performances du programme Apply ne sont pas optimisées si un seul ensemble d'abonnements est traité.</p> <p><b>*YES</b> Les performances du programme Apply sont optimisées si un seul ensemble d'abonnements est traité. Le programme Apply réutilise les informations de l'ensemble d'abonnements pendant les cycles de traitement ultérieurs, exigeant ainsi moins de ressources de l'unité centrale et améliorant le débit.</p>

## Notes sur l'utilisation

Vous pouvez configurer le système pour qu'il lance automatiquement le sous-système en ajoutant la commande mentionnée dans la valeur QSTRUPPGM sur votre système. Si vous utilisez le sous-système QDP4/QZSNDPR, il est lancé dans le cadre du traitement de la commande **STRDPRAPY**.

Si la base de données relationnelle (RDB) spécifiée par le paramètre **CTLSVR** est une base de données DB2 pour i5/OS, les tables sur le serveur se trouvent dans la bibliothèque ASN. Si la base de données relationnelle n'est pas une base de données DB2 pour i5/OS, vous pouvez accéder aux tables à l'aide d'ASN en tant que qualificatif.

## Cas d'erreur lors du démarrage du programme Apply

La commande **STRDPRAPY** émet un message d'erreur si l'une des conditions suivantes se produit :

- Si l'utilisateur n'existe pas.
- Si l'utilisateur qui exécute la commande n'est pas autorisé sur le profil utilisateur spécifié dans la commande ou la description de travail.
- Si une instance du programme Apply est déjà active sur le système local pour cette combinaison du qualificatif Apply et du serveur de contrôle.
- Si le nom RDB spécifié par le paramètre **CTLSVR** n'est pas dans le répertoire de base de données relationnelle.

- Si les tables de contrôle ne quittent pas sur la RDB spécifiée par le paramètre **CTLSVR**.
- Si aucun ensemble d'abonnements n'est défini pour le qualificatif Apply spécifié par le paramètre **APYQUAL**.

Un programme Apply doit être démarré pour chaque qualificatif Apply unique dans chaque table **IBMSNAP\_SUBS\_SET**. Vous pouvez démarrer plusieurs programmes Apply en spécifiant un qualificatif Apply différent chaque fois que vous émettez la commande **STRDPRAPY**. Ces programmes Apply s'exécuteront dans le même profil utilisateur.

## Identification des travaux du programme Apply

Chaque programme Apply est identifié à l'aide du qualificatif Apply et des noms de serveur de contrôle. Lorsqu'il est exécuté, le travail lancé pour le programme Apply ne possède pas suffisamment d'attributs extérieurs pour identifier correctement le programme Apply associé à une combinaison spécifique du qualificatif Apply et du serveur de contrôle. C'est pourquoi le travail est décrit de la manière suivante :

- Le travail est lancé sous le profil utilisateur associé au paramètre **USER**.
- Les dix premiers caractères du qualificatif Apply sont tronqués et deviennent le nom du travail.
- DB2 DataPropagator pour System i conserve une table **IBMSNAP\_APPLY\_JOB** nommé dans la bibliothèque **ASN** sur le système local. La table mappe le qualificatif Apply et les valeurs du serveur de contrôle avec le travail du programme Apply correspondant.
- Vous pouvez voir l'historique de travail. Les noms du qualificatif Apply et du serveur de contrôle sont utilisés dans l'appel du programme Apply.

En général, vous pouvez identifier le travail du programme Apply correspondant en regardant la liste des travaux s'exécutant dans le sous-système **QZSNDPR** si :

- Les dix premiers caractères du nom du qualificatif Apply sont uniques.
- Le programme Apply est démarré uniquement pour le serveur de contrôle local.

## Exemples pour STRDPRAPY

Les exemples suivants illustrent comment utiliser la commande **STRDPRAPY**.

### Exemple 1 :

Pour démarrer le programme Apply qui utilise le qualificatif Apply **AQHR** et les tables de contrôle Apply qui résident localement et pour générer un fichier de trace contenant des informations sur les erreurs et le flux d'exécution :

```
STRDPRAPY APYQUAL(AQHR) CTLSVR(*LOCAL) TRACE(*ALL)
```

### Exemple 2 :

Pour démarrer un programme Apply avec des tables de contrôle Apply qui résident localement et pour indiquer que le travail qui a démarré ce programme Apply se termine automatiquement lorsque le programme Apply n'a plus rien à traiter :

```
STRDPRAPY APYQUAL(AQHR) CTLSVR(*LOCAL) ALWINACT(*NO)
```

### Exemple 3 :

Pour démarrer un programme Apply qui vide la table IBMSNAP\_APPLYTRAIL pendant le démarrage du programme :

```
STRDPRAPY APYQUAL(AQHR) CTLSVR(*LOCAL) TRLREUSE(*YES)
```

#### Exemple 4 :

Pour démarrer un programme Apply avec toutes les valeurs par défaut :

```
STRDPRAPY
```

---

## STRDPRCAP : démarrage de Capture (System i)

Utilisez la commande Start DPR Capture (**STRDPRCAP**) pour commencer à capturer les modifications sur les tables de base de données System i sur les serveurs System i.

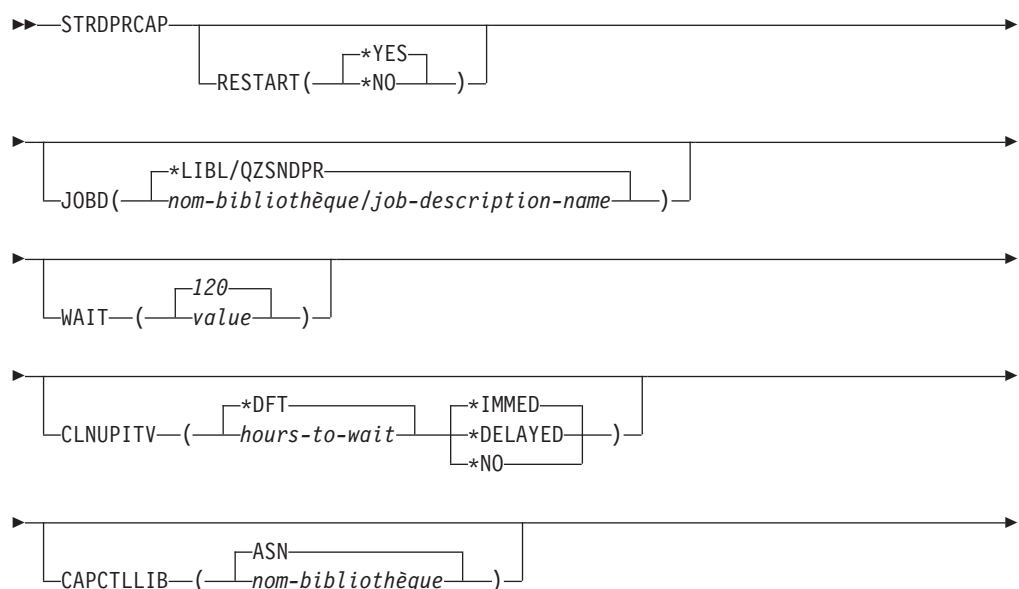
Dans la mesure où cette commande traite toutes les sources de réplication dans la table IBMSNAP\_REGISTER, veuillez à exécuter cette commande en ayant les droits d'accès adéquats.

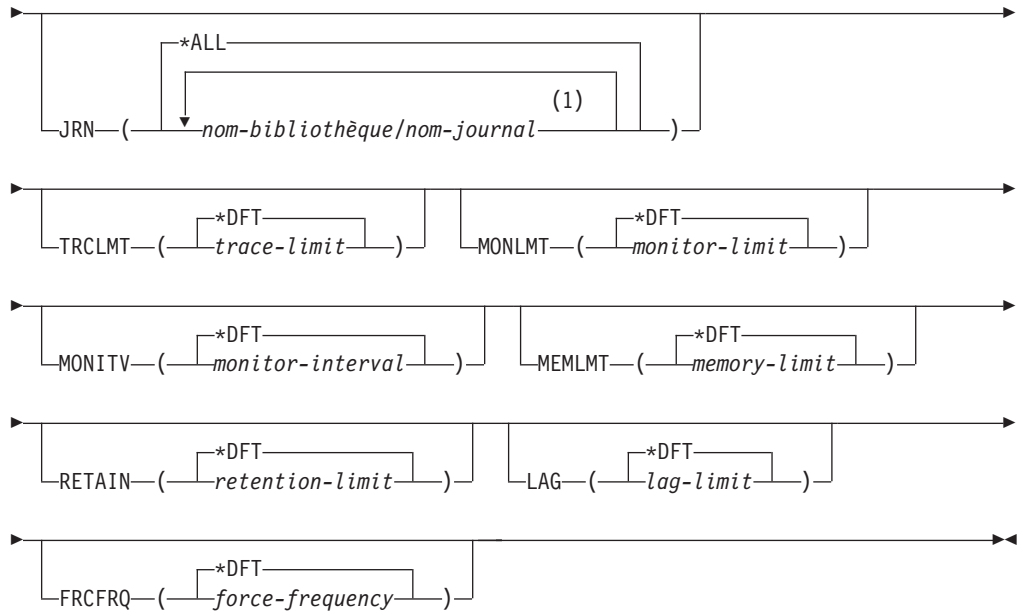
Une fois le programme Capture démarré, il s'exécute en continu jusqu'à ce que vous l'arrêtez ou qu'il détecte une erreur irréparable.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe





**Remarques :**

1 Vous pouvez spécifier un maximum de 50 journaux.

Le tableau 55 dresse la liste des paramètres d'appel.

Tableau 55. Définitions des paramètres de commande STRDPRCAP pour System i

Paramètre	Définition et invites
RESTART	<p>Spécifie comment le programme Capture gère les départs à chaud et à froid.</p> <p><b>*YES (valeur par défaut)</b>            Le programme Capture poursuit le traitement des modifications à partir du point où il se trouvait lorsqu'il s'est arrêté précédemment. C'est ce qu'on appelle également le <i>démarrage à chaud</i> et cela correspond au mode d'opération normal.</p> <p><b>*NO</b>            Le programme Capture supprime toutes les informations des tables de données de modification. Le programme Capture supprime également toutes les informations de la table des unités d'oeuvre (UOW) lorsque vous spécifiez JRN(*ALL).             Tous les abonnements des tables source affectées sont intégralement régénérés avant que la capture des modifications ne recommence. Ce processus est également appelé <i>démarrage à froid</i>.             En spécifiant RESTART(*NO) et JRN(<i>nom-bibliothèque/nom-journal</i>), vous pouvez effectuer un démarrage à froid du programme Capture pour les journaux spécifiés.</p>

Tableau 55. Définitions des paramètres de commande STRDPRCAP pour System i (suite)

Paramètre	Définition et invites
<b>JOB</b>	<p>Spécifie le nom de la description de travail à utiliser lorsque l'on soumet le programme Capture.</p> <p><b>*LIBL/QZSNDPR (valeur par défaut)</b>            Spécifie la description de travail par défaut fournie avec DB2 DataPropagator pour System i.</p> <p><i>nom-bibliothèque/job-description-name</i>            Nom de la description de travail utilisée pour le programme Capture.</p>
<b>WAIT</b>	<p>Spécifie le nombre maximal de secondes (60 à 6 000) à attendre avant que le programme Capture ne vérifie son statut. Vous pouvez utiliser cette valeur pour ajuster la réactivité du programme Capture.</p> <p>Une valeur basse réduit le temps nécessaire au programme Capture avant de se terminer ou de s'initialiser, mais cela peut avoir un effet négatif sur les performances du système. Une valeur supérieure augmente le temps nécessaire au programme Capture avant de se terminer ou de s'initialiser, mais peut améliorer les performances du système. Une valeur qui est trop élevée peut se traduire par une réactivité moins bonne lorsque le programme Capture effectue un traitement régulier. L'ampleur de la baisse de réactivité dépend de l'ampleur de l'activité de modification des tables source et du nombre d'autres travaux exécutés sur le système.</p> <p><b>120 (valeur par défaut)</b>            Le programme Capture attend 120 secondes.</p> <p><i>value</i>            Le nombre maximal de secondes pendant lesquelles le programme Capture attend.</p>

Tableau 55. Définitions des paramètres de commande STRDPRCAP pour System i (suite)

Paramètre	Définition et invites
<b>CLNUPITV</b>	<p>Indique la durée maximale (en heures) avant que le programme Capture n'élague les anciens enregistrements des tables de modification de données (CD), des tables d'unités d'oeuvre (UOW), et des tables IBMSNAP_SIGNAL, IBMSNAP_CAPMON, IBMSNAP_CAPTRACE, et IBMSNAP_AUTHTKN.</p> <p>Ce paramètre fonctionne avec le paramètre <b>RETAIN</b> pour contrôler l'élagage des tables CD, UOW, IBMSNAP_SIGNAL, et IBMSNAP_AUTHTKN, avec le paramètre <b>MONLMT</b> pour contrôler l'élagage de la table IBMSNAP_CAPMON, et avec le paramètre <b>TRCLMT</b> pour contrôler l'élagage de la table IBMSNAP_CAPTRACE.</p> <p>(Utilisez la commande <b>STRDPRCAP</b> pour définir les paramètres <b>RETAIN</b>, <b>MONLMT</b>, et <b>TRCLMT</b> pour le programme Capture. Utilisez la commande <b>CHGDPRCAPA</b> ou <b>OVRDPRCAPA</b> pour modifier la configuration de ces paramètres.)</p> <p>Le paramètre <b>CLNUPITV</b> se compose de deux parties :</p> <p><b>*DFT (valeur par défaut)</b> Le programme Capture utilise la valeur de la colonne PRUNE_INTERVAL de la table IBMSNAP_CAPPARMS.</p> <p><i>hours-to-wait</i> L'intervalle d'élagage exprimé en nombre d'heures spécifique (1 à 100).</p> <p><b>*IMMED (valeur par défaut)</b> Le programme Capture élague les anciens enregistrements au début de l'intervalle spécifié (ou immédiatement), et ensuite à chaque intervalle.</p> <p><b>*DELAYED</b> Le programme Capture élague les anciens enregistrements à la fin de l'intervalle spécifié, et ensuite à chaque intervalle.</p> <p><b>*NO</b> Le programme Capture n'élague pas les enregistrements.</p>
<b>CAPCTLLIB</b>	<p>Spécifie le schéma Capture, qui est le nom de la bibliothèque dans laquelle résident les tables de contrôle Capture.</p> <p><b>ASN (valeur par défaut)</b> La bibliothèque par défaut de la bibliothèque dans laquelle se trouvent les tables de contrôle Capture.</p> <p><i>nom-bibliothèque</i> Le nom de la bibliothèque dans laquelle se trouvent les tables de contrôle Capture.</p>



Tableau 55. Définitions des paramètres de commande STRDPRCAP pour System i (suite)

Paramètre	Définition et invites
JRN	<p>Spécifie un sous-ensemble de 50 journaux maximum avec lequel vous voulez que le programme Capture travaille. Le programme Capture commence le traitement de toutes les tables source actuellement journalisées dans ce journal.</p> <p><b>*ALL (valeur par défaut)</b> Le programme Capture commence à travailler avec tous les journaux dans lesquels sont journalisées des tables source.</p> <p><i>nom-bibliothèque/nom-journal</i> Nom qualifié du journal avec lequel vous souhaitez que le programme Capture travaille. En cas de saisie de plusieurs journaux, séparez les journaux par des espaces.</p>
TRCLMT	<p>Spécifie la limite de trace (en minutes). Le programme Capture élague toute ligne de table IBMSNAP_CAPTRACE qui est plus ancienne que la limite de trace. La valeur par défaut est 10 080 minutes (sept jours d'entrées de trace).</p> <p><b>*DFT (valeur par défaut)</b> Le programme Capture utilise la valeur de la colonne TRACE_LIMIT de la table IBMSNAP_CAPPARMS.</p> <p><i>trace-limit</i> Le nombre de minutes de données de trace conservées dans la table IBMSNAP_CAPTRACE après l'élagage.</p>
MONLMT	<p>Spécifie le nombre maximal de contrôles (en minutes). Le programme Capture élague toute ligne de la table IBMSNAP_CAPMON qui est plus ancienne que le nombre maximal de contrôles. La valeur par défaut est 10 080 minutes (sept jours d'entrées de contrôles).</p> <p><b>*DFT (valeur par défaut)</b> Le programme Capture utilise la valeur de la colonne MONITOR_LIMIT de la table IBMSNAP_CAPPARMS.</p> <p><i>monitor-limit</i> Le nombre de minutes de données de contrôle conservées dans la table IBMSNAP_CAPMON après l'élagage.</p>
MONITV	<p>Indique à quelle fréquence (en secondes) le programme Capture insère des lignes dans la table IBMSNAP_CAPMON. La valeur par défaut est de 300 secondes (cinq minutes).</p> <p><b>*DFT (valeur par défaut)</b> Le programme Capture utilise la valeur de la colonne MONITOR_INTERVAL de la table IBMSNAP_CAPPARMS.</p> <p><i>monitor-interval</i> Le nombre de secondes entre l'insertion de ligne dans la table IBMSNAP_CAPMON. L'intervalle de contrôle doit être d'au moins 120 secondes (deux minutes). Si vous saisissez un nombre inférieur à 120, cette valeur de paramètre est définie sur 120.</p>

Tableau 55. Définitions des paramètres de commande STRDPRCAP pour System i (suite)

Paramètre	Définition et invites
MEMLMT	<p>Spécifie la taille maximale (en mégaoctets) de mémoire que le travail du journal Capture peut utiliser. La valeur par défaut est de 32 mégaoctets.</p> <p><b>*DFT (valeur par défaut)</b>                      Le programme Capture utilise la valeur de la colonne MEMORY_LIMIT de la table IBMSNAP_CAPPARMS.</p> <p><i>memory-limit</i>                      Le nombre maximal de mégaoctets pour la mémoire.</p>
RETAIN	<p>Indique la durée de conservation, qui correspond au nombre de minutes pendant lesquelles les données sont conservées dans la table de modification des données (CD), la table des unités d'oeuvre (UOW), les tables IBMSNAP_SIGNAL, et IBMSNAP_AUTHTKN avant d'être supprimées. Cette valeur fonctionne avec la valeur de paramètre CLNUPITV. Lorsque la valeur CLNUPITV est atteinte, des données CD, UOW, IBMSNAP_SIGNAL, et IBMSNAP_AUTHTKN sont supprimées si ces données dépassent la limite de conservation.</p> <p>Veillez à ce que les intervalles Apply soient définis afin de copier les informations modifiées avant que les données n'atteignent cette valeur de paramètre RETAIN, et ce, afin d'éviter des données incohérentes dans vos tables. Si les données deviennent incohérentes, le programme Apply effectue une régénération intégrale.</p> <p>La valeur par défaut est de 10 080 minutes (sept jours). Le maximum est de 35000000 minutes.</p> <p><b>*DFT (valeur par défaut)</b>                      Le programme Capture utilise la valeur de la colonne RETENTION_LIMIT de la table IBMSNAP_CAPPARMS.</p> <p><i>retention-limit</i>                      Le nombre de minutes pendant lesquelles les données CD, UOW, IBMSNAP_SIGNAL, et IBMSNAP_AUTHTKN sont conservées.</p>
LAG	<p>Spécifie la nouvelle limite de décalage, qui correspond au nombre de minutes pendant lesquelles le programme Capture peut subir un retard de traitement avant de redémarrer.</p> <p>Lorsque la limite de décalage est atteinte (à savoir, lorsque l'horodatage de l'entrée de journal est antérieur à l'horodatage actuel moins la limite de décalage), le programme Capture lance un démarrage à froid pour les tables qu'il traite dans ce journal. Le programme Apply effectue ensuite une régénération intégrale afin de fournir au programme Capture un nouveau point de départ.</p> <p>La valeur par défaut est de 10 080 minutes (sept jours). Le maximum est de 35000000 minutes.</p> <p><b>*DFT (valeur par défaut)</b>                      Le programme Capture utilise la valeur de la colonne LAG_LIMIT de la table IBMSNAP_CAPPARMS.</p> <p><i>lag-limit</i>                      Le nombre de minutes de retard autorisées pour le programme Capture.</p>

Tableau 55. Définitions des paramètres de commande **STRDPRCAP** pour System i (suite)

Paramètre	Définition et invites
<b>FRCFRQ</b>	<p>Indique la fréquence (de 30 à 600 secondes) selon laquelle le programme Capture écrit les modifications dans les tables de données de modification (CD) et d'unités d'oeuvre (UOW). Le programme Capture met ces modifications à la disposition du programme Apply soit lorsque les mémoires tampons sont remplies, soit lorsque la limite de temps FRCFRQ expire, quel que soit le plus tôt.</p> <p>Utilisez ce paramètre pour rendre les modifications immédiatement disponibles pour le programme Apply sur les serveurs avec un faible taux de modifications de table source. La valeur de paramètre <b>FRCFRQ</b> est une valeur globale utilisée pour toutes les tables sources définies. La configuration de la valeur <b>FRCFRQ</b> sur un faible nombre peut affecter les performances du système.</p> <p>La valeur par défaut est de 30 secondes.</p> <p><b>*DFT (valeur par défaut)</b>            Le programme Capture utilise la valeur de la colonne COMMIT_INTERVAL de la table IBMSNAP_CAPPARMS.</p> <p><i>force-frequency</i>            Le nombre de secondes pendant lesquelles le programme Capture conserve les modifications de la table de données de modification et des unités d'oeuvre dans l'espace de la mémoire tampon avant de mettre ces modifications à la disposition du programme Apply.</p>

## Notes sur l'utilisation

Le paramètre **CLNUPITV** de la commande **STRDPRCAP** indique le nombre maximal d'heures pendant lesquelles le programme Capture attend avant d'élaguer les anciens enregistrements des tables de données de modification (CD), des unités d'oeuvre (UOW), **IBMSNAP\_SIGNAL**, **IBMSNAP\_CAPMON**, **IBMSNAP\_CAPTRACE**, et **IBMSNAP\_AUTHTKN**.

Vous pouvez exécuter la commande **STRDPRCAP** manuellement, ou vous pouvez exécuter automatiquement la commande, dans le cadre du démarrage du système (programme de démarrage IPL).

Si la description du travail spécifié avec le paramètre **JOBID** utilise la file d'attente des travaux QDP4/QZSNDPR et que le sous-système DB2 DataPropagator pour System i n'est pas actif, la commande **STRDPRCAP** lance le sous-système. Si la description du travail est définie pour utiliser une file d'attente de travaux et un sous-système différents, vous devez démarrer manuellement ce sous-système avec la commande Start Subsystem (**STRSBS**) soit avant soit après avoir exécuté la commande **STRDPRCAP** :

```
STRSBS QDP4/QZSNDPR
```

Vous pouvez configurer le système pour qu'il lance automatiquement le sous-système en ajoutant la commande **STRSBS** dans le programme mentionné dans la valeur système **QSTRUPPGM** sur votre système.

## Redémarrage de Capture à l'aide de démarrages à chaud ou à froid

La valeur du paramètre **RESTART** dans la commande **STRDPRCAP** contrôle la manière dont le programme Capture gère les démarrages à chaud et à froid.

### Processus de démarrage à chaud

Dans la plupart des cas, les informations de démarrage à chaud sont sauvegardées. De manière occasionnelle, ce n'est pas le cas. Dans cette situation, le programme Capture utilise les tables CD, la table UOW, ou la table IBMSNAP\_PRUNCNTL pour se resynchroniser sur l'heure à laquelle il a été arrêté.

### Démarrages à froid automatiques

Parfois, le programme Capture bascule automatiquement sur un démarrage à froid, même si vous avez spécifié un démarrage à chaud. Sur les systèmes System i, les démarrages à froid fonctionnent sur une base journal par journal. Ainsi, par exemple, si un journal dépasse la limite de décalage, toutes les sources de réplication qui utilisent ce journal sont démarrées à froid, tandis que les sources de réplication qui utilisent un autre journal ne le sont pas.

## Exemples pour STRDPRCAP

Les exemples suivants illustrent comment utiliser la commande **STRDPRCAP**.

### Exemple 1 :

Pour lancer un démarrage à chaud d'un programme Capture pour deux journaux différents :

```
STRDPRCAP RESTART(*YES) JRN(HR/QSQJRN ACCTS/QSQJRN)
```

### Exemple 2 :

Pour démarrer un programme Capture pour un journal spécifié :

```
STRDPRCAP CAPCTLLIB(BSN) JRN(MARKETING/QSQJRN)
```

Les tables de contrôle Capture résident dans une bibliothèque nommée BSN.

### Exemple 3 :

Pour démarrer un programme Capture sans élagage pour deux journaux :

```
STRDPRCAP RESTART(*YES) CLNUPITV(*DFT *NO) JRN(HR/QSQJRN ACCTS/QSQJRN)
```

### Exemple 4 :

Pour démarrer un programme Capture pour un journal spécifié dans la bibliothèque de contrôle Capture par défaut et pour modifier l'élagage de limite de trace par défaut, l'élagage de la limite de contrôle, l'insertion de la table IBMSNAP\_CAPMON, et les paramètres de limite de mémoire :

```
STRDPRCAP CAPCTLLIB(ASN) JRN(SALES/QSQJRN) TRCLMT(1440) MONLMT(1440)  
MONITV(3600) MEMLMT(64)
```

### Exemple 5 :

Pour lancer un démarrage à froid d'un programme Capture :

## WRKDPTRTC : utilisation de la fonction trace DPR (System i)

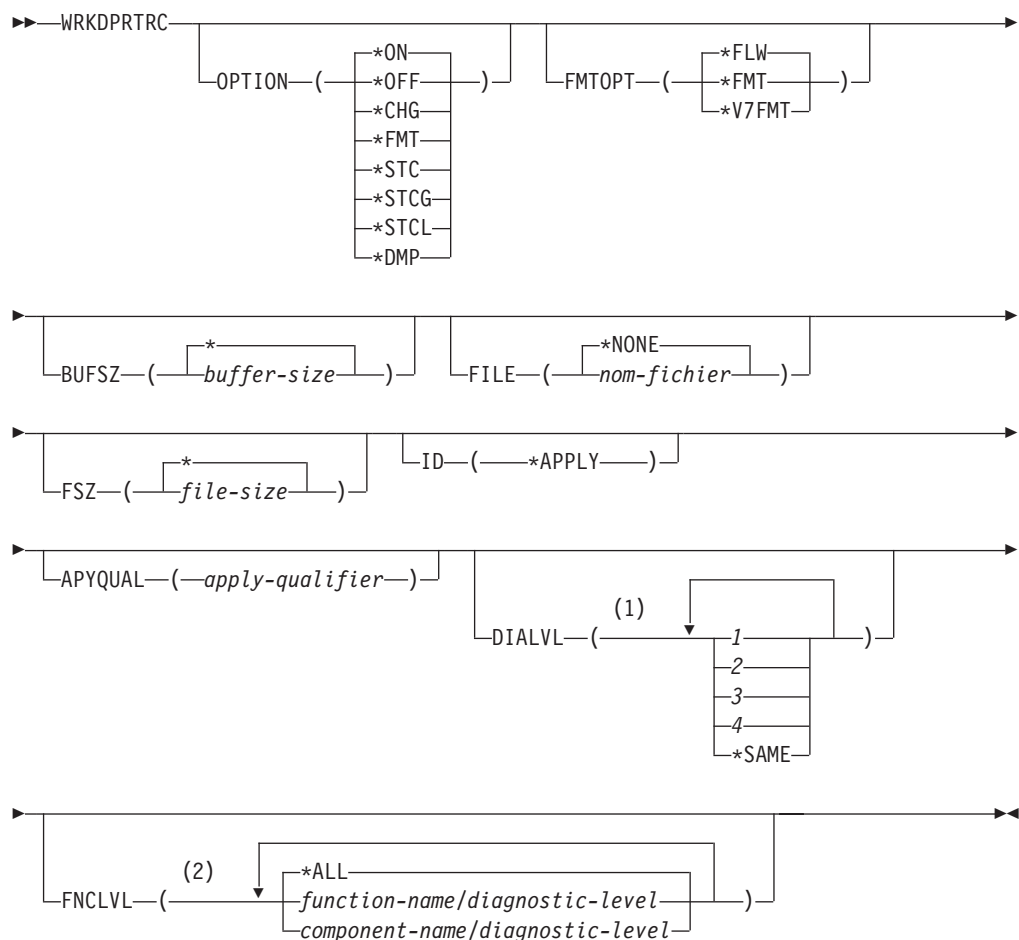
### System i

N'utilisez la commande DPR trace (**WRKDPTRTC**) que si le support logiciel IBM vous a demandé d'utiliser la commande. La commande exécute la fonction trace pour consigner les informations du flux de programmes pour les programmes Apply spécifiés.

Après avoir saisi le nom de commande dans la ligne de commande, vous pouvez appuyer sur la touche F4 pour afficher la syntaxe de la commande.

Pour afficher une description complète de cette commande et de tous ses paramètres, déplacez le curseur sur la commande en haut de l'écran et appuyez sur la touche F1. Pour afficher une description d'un paramètre spécifique, placez le curseur sur ce paramètre et appuyez sur la touche F1.

### Syntaxe



### Remarques :

- 1 Vous pouvez spécifier plusieurs valeurs.
- 2 Vous pouvez spécifier jusqu'à 20 fonctions ou composants.

Le tableau 56 dresse la liste des paramètres d'appel.

Tableau 56. Définitions des paramètres de commande *WRKDPTRTC* pour System i

Paramètre	Définition
<b>OPTION</b>	<p>Spécifie une fonction trace.</p> <p><b>*ON (valeur par défaut)</b> Active la fonction trace. Cette option crée automatiquement un segment de mémoire partagée pour le traçage.</p> <p><b>*OFF</b> Désactive la fonction trace.</p> <p><b>*CHG</b> Modifie les valeurs des paramètres de la fonction trace.</p> <p><b>*FMT</b> Formate la sortie de la fonction trace depuis la mémoire partagée.</p> <p><b>*STC</b> Affiche le statut d'une fonction trace. Ces informations sur le statut comprennent 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 de statut, et l'horodatage du programme.</p> <p>Cette option de paramètre est équivalente à l'option stat de la commande <b>asntrc</b> utilisée sur les systèmes d'exploitation UNIX, Windows, et z/OS.</p> <p><b>*STCG</b> Affiche le statut d'une fonction trace dans un format lisible par le centre de réplication.</p> <p><b>*STCL</b> Affiche le statut d'une fonction trace avec des informations supplémentaires sur le niveau de version. Les informations supplémentaires comprennent les niveaux de service de chaque module de l'application et apparaissent comme une longue chaîne de texte.</p> <p>Cette option de paramètre est équivalente à l'option statlong de la commande <b>asntrc</b> utilisé sur les systèmes d'exploitation UNIX, Windows, et z/OS.</p> <p><b>*DMP</b> Ecrit le contenu actuel de la mémoire tampon de trace dans un fichier.</p> <p>Lors de l'invite sur la commande <b>WRKDPTRTC</b>, vous pouvez appuyer sur la touche F4 pour afficher une liste des options de trace.</p>

Tableau 56. Définitions des paramètres de commande WRKDPRTTC pour System i (suite)

Paramètre	Définition
FMTOPT	<p>Spécifie les options de l'ID de format et est utilisée avec le paramètre <b>OPTION(*FMT)</b>.</p> <p><b>*FLW (valeur par défaut)</b> Affiche le flux des appels de fonction.</p> <p><b>*FMT</b> Affiche le format de la mémoire tampon de trace ou du fichier de trace. Affiche toutes les données détaillées.</p> <p><b>*V7FMT</b> Formate les informations de la mémoire tampon de trace ou du fichier de trace au format de la Version 7.</p> <p>Lors de l'invite sur la commande <b>WRKDPRTTC</b>, vous pouvez appuyer sur la touche F4 pour afficher une liste des options de format.</p>
BUFSZ	<p>Indique la taille (en octets) de la mémoire tampon de trace. Vous pouvez saisir M, K, ou G après le chiffre pour indiquer mégaoctets, kilooctets, ou gigaoctets.</p> <p>La valeur par défaut est de deux mégaoctets.</p> <p><b>*INTERVAL (valeur par défaut)</b> Utilise la taille par défaut de deux mégaoctets.</p> <p><i>buffer-size</i> La taille de la mémoire tampon en octets.</p>
FILE	<p>Indique si la sortie de trace est écrite dans un fichier.</p> <p><b>*NONE (valeur par défaut)</b> La sortie de trace va uniquement dans la mémoire partagée.</p> <p><i>nom-fichier</i> Le nom du fichier de sortie. Si vous utilisez le paramètre <b>OPTION(*DMP)</b>, ce nom de fichier représente le nom d'un fichier de visage.</p>
FSZ	<p>Indique la taille (en octets) du fichier dans lequel sont stockées les données de trace. Vous pouvez saisir M, K, ou G après le chiffre pour indiquer mégaoctets, kilooctets, ou gigaoctets.</p> <p>La valeur par défaut est de deux gigaoctets.</p> <p><b>* (valeur par défaut)</b> Utilise la taille par défaut de deux gigaoctets.</p> <p><i>file-size</i> La taille du fichier en octets.</p>
ID	<p>Indique le type de programme à tracer.</p> <p><b>*APPLY (valeur par défaut)</b> Une trace du programme Apply.</p>
APYQUAL	<p>Indique le nom du programme Apply à tracer.</p> <p><i>apply-qualifier</i> Nom du qualificatif Apply.</p>

Tableau 56. Définitions des paramètres de commande WRKDPRTTC pour System i (suite)

Paramètre	Définition
DIALVL	<p>Indique les types d'enregistrement de trace à enregistrer par la fonction de trace. Les enregistrements de trace sont classés par numéro de masque de diagnostic :</p> <ol style="list-style-type: none"> <li>1 Données de flux, qui comprennent les points d'entrée et de sortie des fonctions.</li> <li>2 Données de base, qui comprennent tous les événements majeurs rencontrés par la fonction trace.</li> <li>3 Données détaillées, qui comprennent les événements majeurs et leurs descriptions.</li> <li>4 Données de performance.</li> </ol> <p><b>*SAME</b>            Cette commande utilise les paramètres de niveau de diagnostic à partir de la précédente fonction trace.</p> <p>Vous pouvez saisir un ou plusieurs numéros de masque de diagnostic. Les numéros que vous saisissez doivent être dans l'ordre croissant. Ne saisissez pas d'espaces entre les chiffres.</p> <p><b>Important :</b> Les niveaux des numéros ne sont <i>pas</i> inclus.</p> <p>Lorsque vous lancez la fonction trace, la valeur par défaut est DIALVL(1234). Lorsque vous appelez ensuite la fonction trace, la valeur par défaut est *SAME.</p> <p>Lors de l'invite sur la commande <b>WRKDPRTTC</b>, vous pouvez appuyer sur la touche F4 pour afficher une liste des niveaux de diagnostic disponibles.</p>
FNCLVL	<p>Indique si un identifiant de fonction ou d'un composant spécifique doit être tracé.</p> <p><b>*ALL (valeur par défaut)</b>            Toutes les fonctions et composants sont inclus dans la fonction trace.</p> <p><i>function-name/diagnostic-level</i>            Le nom de la fonction à tracer et les numéros de masque de diagnostic correspondants.</p> <p><i>component-name/diagnostic-level</i>            Le nom du composant à tracer et les numéros de masque de diagnostic correspondants.</p> <p>Vous pouvez saisir jusqu'à 20 noms de fonction ou de composant.</p>

## Exemples pour WRKDPRTTC

Les exemples suivants illustrent la manière d'utiliser la commande **WRKDPRTTC**.

### Exemple 1 :



Pour lancer une trace Apply sur le qualificatif Apply AQ1 pour toutes les fonctions et composants avec une sortie écrite dans un fichier appelé TRCFILE :

```
WRKDPTRTC OPTION(*ON) FILE(TRCFILE) ID(*APPLY) APYQUAL(AQ1)
```

**Exemple 2 :**

Pour arrêter une trace Apply sur le qualificatif Apply AQ1 :

```
WRKDPTRTC OPTION(*OFF) ID(*APPLY) APYQUAL(AQ1)
```

**Exemple 3 :**

Pour modifier une trace Apply sur le qualificatif Apply AQ1 sur les niveaux de diagnostic 3 et 4 (données détaillées et données de performance) pour toutes les fonctions et composants :

```
WRKDPTRTC OPTION(*CHG) ID(*APPLY) APYQUAL(AQ1) DIALVL(34)
```

**Exemple 4 :**

Pour afficher le statut d'une trace Apply sur le qualificatif Apply AQ1 :

```
WRKDPTRTC OPTION(*STC) ID(*APPLY) APYQUAL(AQ1)
```

**Exemple 5 :**

Pour afficher les appels de fonction sur le qualificatif Apply AQ1 aux niveaux de diagnostic 3 et 4 :

```
WRKDPTRTC OPTION(*FMT) FMTOPT(*FLW) ID(*APPLY) APYQUAL(AQ1) DIALVL (34)
```

**Exemple 6 :**

Pour écrire les informations de trace Apply du qualificatif Apply AQ1 dans un fichier de vidage nommé DMPFILE :

```
WRKDPTRTC OPTION(*DMP) FILE(DMPFILE) ID(*APPLY) APYQUAL(AQ1)
```



---

## Chapitre 23. Structures des tables de réplication SQL

Les tables de bases de données relationnelles permettent de stocker des informations pour le programme de réplication sur chaque serveur : serveur de contrôle de Capture, serveur de contrôle Apply, serveur de contrôle de Monitor et serveur cible. Ces tables sont appelées *tables de contrôle*.

---

### Tables d'aperçu rapide

Les diagrammes suivants peuvent être utilisés comme guide de référence dans les tables de contrôle du serveur de contrôle Capture, le serveur de contrôle Apply et le serveur de contrôle Monitor.

La figure 7, à la page 408, la figure 8, à la page 409 et la figure 9, à la page 410 affichent les tables du serveur de contrôle Capture, ainsi que les colonnes et les index de chaque table. La figure 10, à la page 411 et la figure 11, à la page 412 affichent les tables et le serveur de contrôle Apply, ainsi que les colonnes et les index de chaque table. La figure 12, à la page 413 et la figure 13, à la page 414 affichent les tables du serveur de contrôle Monitor, ainsi que les colonnes et les index de chaque table.

**Tables de contrôle utilisées dans le serveur de contrôle Capture (image 1 de 2)**

<p><i>OS/400 uniquement</i></p> <p><b>schéma.IBMSNAP_AUHTKN</b> (pas d'index unique)</p> <p>APPLY_QUAL CHAR(18) NOT NULL            IBMSNAP_AUHTKN CHAR(26) NOT NULL            JRN_LIB CHAR(10) NOT NULL            JRN_NAME CHAR(10) NOT NULL            IBMSNAP_LOGMARKER TIMESTAMP NOT NULL</p>	<p><b>ASN.IBMSNAP_CAPSCHEMAS</b> (CAP_SCHEMA_NAME)</p> <p>CAP_SCHEMA_NAME VARCHAR(30)</p> <p><i>OS/400</i></p> <p><b>ASN.IBMSNAP_CAPSCHEMAS</b> (CAP_SCHEMA_NAME)</p> <p>CAP_SCHEMA_NAME VARCHAR(30)            STATUS CHAR(1)</p>
<p><i>UNIX, Windows et z/OS uniquement</i></p> <p><b>schéma.IBMSNAP_CAPENQ</b> (pas d'index unique)</p> <p>LOCKNAME CHAR(9)</p>	<p><b>schéma.IBMSNAP_CAPTRACE</b> (TRACE_TIME)</p> <p>OPERATION CHAR(8) NOT NULL            TRACE_TIME TIMESTAMP NOT NULL            DESCRIPTION VARCHAR(1024) NOT NULL</p> <p><i>OS/400</i></p> <p><b>schéma.IBMSNAP_CAPTRACE</b> (TRACE_TIME)</p> <p>OPERATION CHAR(8) NOT NULL            TRACE_TIME TIMESTAMP NOT NULL            JOB_NAME CHAR(26) NOT NULL            JOB_STR_TIME TIMESTAMP NOT NULL            DESCRIPTION VARCHAR(298) NOT NULL</p>
<p><b>schéma.IBMSNAP_CAPMON</b> (MONITOR_TIME)</p> <p>MONITOR_TIME TIMESTAMP NOT NULL            RESTART_TIME TIMESTAMP NOT NULL            CURRENT_MEMORY INT NOT NULL            CD_ROWS_INSERTED INT NOT NULL            RECAP_ROWS_SKIPPED INT NOT NULL            TRIGR_ROWS_SKIPPED INT NOT NULL            CHG_ROWS_SKIPPED INT NOT NULL            TRANS_PROCESSED INT NOT NULL            TRANS_SPILLED INT NOT NULL            MAX_TRAN_SIZE INT NOT NULL            LOCKING_RETRIES INT NOT NULL            JRN_LIB CHAR(10)            JRN_NAME CHAR(10)            LOGREADLIMIT INT NOT NULL            CAPTURE_IDLE INT NOT NULL            SYNCHTIME TIMESTAMP NOT NULL</p>	<p><b>schéma.IBMSNAP_PRUNCNTL</b> (SOURCE_OWNER, SOURCE_TABLE, SOURCE_VIEW_QUAL, APPLY_QUAL, SET_NAME, TARGET_SERVER, TARGET_TABLE, TARGET_OWNER, MAP_ID)</p> <p>TARGET_SERVER CHAR(18) NOT NULL            TARGET_OWNER VARCHAR(30) NOT NULL            TARGET_TABLE VARCHAR(128) NOT NULL            SYNCHTIME TIMESTAMP            SYNCHPOINT CHAR(10) FOR BIT DATA            SOURCE_OWNER VARCHAR(30) NOT NULL            SOURCE_TABLE VARCHAR(128) NOT NULL            SOURCE_VIEW_QUAL SMALLINT NOT NULL            APPLY_QUAL CHAR(18) NOT NULL            SET_NAME CHAR(18) NOT NULL            CNTL_SERVER CHAR(18) NOT NULL            TARGET_STRUCTURE SMALLINT NOT NULL            CNTL_ALIAS CHAR(8)            PHYS_CHANGE_OWNER VARCHAR(30)            PHYS_CHANGE_TABLE VARCHAR(128)            MAP_ID VARCHAR(10) NOT NULL</p>
<p><b>schéma.IBMSNAP_CAPPARMS</b> (pas d'index unique)</p> <p>RETENTION_LIMIT INT            LAG_LIMIT INT            COMMIT_INTERVAL INT            PRUNE_INTERVAL INT            TRACE_LIMIT INT            MONITOR_LIMIT INT            MONITOR_INTERVAL INT            MEMORY_LIMIT SMALLINT            REMOTE_SRC_SERVER CHAR(18)            AUTOPRUNE CHAR(1)            TERM CHAR(1)            AUTOSTOP CHAR(1)            LOGREUSE CHAR(1)            LOGSDOUT CHAR(1)            SLEEPINTERVAL SMALLINT            CAPTURE_PATH VARCHAR(1040)            STARTMODE VARCHAR(10)</p>	<p><b>schéma.IBMSNAP_PRUNE_LOCK</b> (pas d'index unique)</p> <p>DUMMY CHAR(1)</p>

*Figure 7. Tables utilisées dans le serveur de contrôle Capture. Ces tables sont utilisées par le programme Capture, le programme Apply et les déclencheurs Capture dans le serveur de contrôle de Capture. Les colonnes constituant l'index principal de chaque table sont affichées entre parenthèses sous le nom de la table.*

## Tables de contrôle utilisées dans le serveur de contrôle Capture (image 2 de 2)

<b>schéma.IBMSNAP_PRUNE_SET</b> (TARGET_SERVER, APPLY_QUAL, SET_NAME) TARGET_SERVER CHAR(18) NOT NULL APPLY_QUAL CHAR(18) NOT NULL SET_NAME CHAR(18) NOT NULL SYNCHTIME TIMESTAMP SYNCHPOINT CHAR(10) FOR BIT DATA NOT NULL	<b>schéma.IBMSNAP_REG_SYNCH</b> (TRIGGER_ME) TRIGGER_ME CHAR(1) NOT NULL
<b>schéma.IBMSNAP_REGISTER</b> (SOURCE_OWNER, SOURCE_TABLE, SOURCE_VIEW_QUAL) SOURCE_OWNER VARCHAR(30) NOT NULL SOURCE_TABLE VARCHAR(128) NOT NULL SOURCE_VIEW_QUAL SMALLINT NOT NULL GLOBAL_RECORD CHAR(1) NOT NULL SOURCE_STRUCTURE SMALLINT NOT NULL SOURCE_CONDENSED CHAR(1) NOT NULL SOURCE_COMPLETE CHAR(1) NOT NULL CD_OWNER VARCHAR(30) CD_TABLE VARCHAR(128) PHYS_CHANGE_OWNER VARCHAR(30) PHYS_CHANGE_TABLE VARCHAR(128) CD_OLD_SYNCHPOINT CHAR(10) FOR BIT DATA CD_NEW_SYNCHPOINT CHAR(10) FOR BIT DATA DISABLE_REFRESH SMALLINT NOT NULL CCD_OWNER VARCHAR(30) CCD_TABLE VARCHAR(128) CCD_OLD_SYNCHPOINT CHAR(10) FOR BIT DATA SYNCHPOINT CHAR(10) FOR BIT DATA SYNCHTIME TIMESTAMP CCD_CONDENSED CHAR(1) CCD_COMPLETE CHAR(1) ARCH_LEVEL CHAR(4) NOT NULL DESCRIPTION CHAR(254) BEFORE_IMG_PREFIX VARCHAR(4) CONFLICT_LEVEL CHAR(1) CHG_UPD_TO_DEL_INS CHAR(1) CHGONLY CHAR(1) RECAPTURE CHAR(1) OPTION_FLAGS CHAR(4) NOT NULL STOP_ON_ERROR CHAR(1) STATE CHAR(1) STATE_INFO CHAR(8)	<b>schéma.IBMSNAP_RESTART</b> (pas d'index unique) MAX_COMMITSEQ CHAR(10) FOR BIT DATA NOT NULL MAX_COMMIT_TIME TIMESTAMP NOT NULL MIN_INFLIGHTSEQ CHAR(10) FOR BIT DATA NOT NULL CURR_COMMIT_TIME TIMESTAMP NOT NULL CAPTURE_FIRST_SEQ CHAR(10) FOR BIT DATA NOT NULL OS/400 <b>schéma.IBMSNAP_RESTART</b> (JRN_LIB, JRN_NAME) MAX_COMMITSEQ CHAR(10) FOR BIT DATA NOT NULL MAX_COMMIT_TIME TIMESTAMP NOT NULL MIN_INFLIGHTSEQ CHAR(10) FOR BIT DATA NOT NULL CURR_COMMIT_TIME TIMESTAMP NOT NULL CAPTURE_FIRST_SEQ CHAR(10) FOR BIT DATA NOT NULL UID INTEGER NOT NULL SEQNBR BIGINT NOT NULL JRN_LIB CHAR(10) NOT NULL JRN_NAME CHAR(10) NOT NULL STATUS CHAR(1)
OS/400 uniquement <b>schéma.IBMSNAP_REG_EXT</b> (VERSION, SOURCE_OWNER, SOURCE_TABLE, SOURCE_VIEW_QUAL) VERSION INT NOT NULL SOURCE_OWNER VARCHAR(30) NOT NULL SOURCE_TABLE VARCHAR(128) NOT NULL SOURCE_NAME CHAR(10) SOURCE_MBR CHAR(10) SOURCE_TABLE_RDB CHAR(18) JRN_LIB CHAR(10) JRN_NAME CHAR(10) FR_START_TIME TIMESTAMP SOURCE_VIEW_QUAL SMALLINT NOT NULL CMT_BEHAVIOR_CASE SMALLINT NOT NULL WITH DEFAULT MAX_ROWS_BTWN_CMTS SMALLINT NOT NULL WITH DEFAULT	<b>schéma.IBMSNAP_SEQTABLE</b> (SEQ) SEQ INTEGER NOT NULL
	<b>schéma.IBMSNAP_SIGNAL</b> (SIGNAL_TIME) SIGNAL_TIME TIMESTAMP NOT NULL WITH DEFAULT SIGNAL_TYPE VARCHAR(30) NOT NULL SIGNAL_SUBTYPE VARCHAR(30) SIGNAL_INPUT_IN VARCHAR(500) SIGNAL_STATE CHAR(1) NOT NULL SIGNAL_LSN CHAR(10) FOR BIT DATA
	<b>schéma.IBMSNAP_UOW</b> (IBMSNAP_COMMITSEQ, IBMSNAP_LOGMARKER) IBMSNAP_UOWID CHAR(10) FOR BIT DATA NOT NULL IBMSNAP_COMMITSEQ CHAR(10) FOR BIT DATA NOT NULL IBMSNAP_LOGMARKER TIMESTAMP NOT NULL IBMSNAP_AUTHTKN VARCHAR(30) NOT NULL IBMSNAP_AUTHID VARCHAR(30) NOT NULL IBMSNAP_REJ_CODE CHAR(1) NOT NULL WITH DEFAULT IBMSNAP_APPLY_QUAL CHAR(18) NOT NULL WITH DEFAULT

Figure 8. Tables utilisées dans le serveur de contrôle Capture (suite). Ces tables sont utilisées par le programme Capture, le programme Apply et les déclencheurs Capture dans le serveur de contrôle Capture. Les colonnes constituant l'index principal de chaque table sont affichées entre parenthèses sous le nom de la table.

## Tables de contrôle utilisées dans le serveur de contrôle Capture (image 3 de 3)

### schéma.IBMSNAP\_RESTART

<i>UNIX, Windows et z/OS uniquement</i> (pas d'index)	
MAX_COMMITSEQ	CHAR(10) FOR BIT DATA NOT NULL
MAX_COMMIT_TIME	TIMESTAMP NOT NULL
MIN_INFLIGHTSEQ	CHAR(10) FOR BIT DATA NOT NULL
CURR_COMMIT_TIME	TIMESTAMP NOT NULL
CAPTURE_FIRST_SEQ	CHAR(10) FOR BIT DATA NOT NULL
<i>OS/400 uniquement</i> (JRN_LIB, JRN_NAME)	
MAX_COMMITSEQ	CHAR(10) FOR BIT DATA NOT NULL
MAX_COMMIT_TIME	TIMESTAMP NOT NULL
MIN_INFLIGHTSEQ	CHAR(10) FOR BIT DATA NOT NULL
CURR_COMMIT_TIME	TIMESTAMP NOT NULL
CAPTURE_FIRST_SEQ	CHAR(10) FOR BIT DATA NOT NULL
UID	INTEGER NOT NULL
SEQNBR	BIGINT NOT NULL
JRN_LIB	CHAR(10) NOT NULL
JRN_NAME	CHAR(10) NOT NULL
STATUS	CHAR(1)

### schéma.IBMSNAP\_SIGNAL

SIGNAL_TIME	TIMESTAMP NOT NULL WITH DEFAULT
SIGNAL_TYPE	VARCHAR(30) NOT NULL
SIGNAL_SUBTYPE	VARCHAR(30)
SIGNAL_INPUT_IN	VARCHAR(500)
SIGNAL_STATE	CHAR(1) NOT NULL
SIGNAL_LSN	CHAR(10) FOR BIT DATA

### schéma.IBMSNAP\_UOW

(IBMSNAP_COMMITSEQ, IBMSNAP_LOGMARKER)	
IBMSNAP_UOWID	CHAR(10) FOR BIT DATA NOT NULL
IBMSNAP_COMMITSEQ	CHAR(10) FOR BIT DATA NOT NULL
IBMSNAP_LOGMARKER	TIMESTAMP NOT NULL
IBMSNAP_AUTHTKN	VARCHAR(30) NOT NULL
IBMSNAP_AUTHID	VARCHAR(30) <sup>1</sup> NOT NULL
IBMSNAP_REJ_CODE	CHAR(1) NOT NULL WITH DEFAULT
IBMSNAP_APPLY_QUAL	CHAR(18) NOT NULL WITH DEFAULT

### schéma.IBMSNAP\_SEQTABLE

<i>Informix uniquement</i> (SEQ)	
SEQ	INTEGER NOT NULL

<sup>1</sup> VARCHAR(30) pour DB2 pour z/OS, version 8 ou précédente, mode de compatibilité ; VARCHAR(128) pour DB2 pour z/OS, version 8, mode nouvelle fonction.

<sup>2</sup> VARCHAR(18) pour DB2 pour z/OS, version 8 ou précédente, mode de compatibilité ; VARCHAR(128) pour DB2 pour z/OS, version 8, mode nouvelle fonction.

Figure 9. Tables utilisées dans le serveur de contrôle Capture (suite). Ces tables sont utilisées par le programme Capture, le programme Apply et les déclencheurs Capture dans le serveur de contrôle de Capture. Les colonnes constituant l'index principal de chaque table sont affichées entre parenthèses sous le nom de la table.

## Tables de contrôle utilisées dans le serveur de contrôle Apply (image 1 de 2)

<b>ASN.IBMSNAP_APPLYTRAIL</b> (pas d'index unique)		<b>ASN.IBMSNAP_APPENQ</b> (APPLY_QUAL)	
APPLY_QUAL	CHAR(18) NOT NULL	APPLY_QUAL	CHAR(18)
SET_NAME	CHAR(18) NOT NULL		
SET_TYPE	CHAR(1) NOT NULL		
WHOS_ON_FIRST	CHAR(1) NOT NULL		
ASNLOAD	CHAR(1)		
FULL_REFRESH	CHAR(1)		
EFFECTIVE_MEMBERS	INT		
SET_INSERTED	INT NOT NULL		
SET_DELETED	INT NOT NULL		
SET_UPDATED	INT NOT NULL		
SET_REWORKED	INT NOT NULL		
SET_REJECTED_TRXS	INT NOT NULL		
STATUS	SMALLINT NOT NULL		
LASTRUN	TIMESTAMP NOT NULL		
LASTSUCCESS	TIMESTAMP		
SYNCHPOINT	CHAR(10) FOR BIT DATA		
SYNCHTIME	TIMESTAMP		
SOURCE_SERVER	CHAR (18) NOT NULL		
SOURCE_ALIAS	CHAR(8)		
SOURCE_OWNER	VARCHAR(30)		
SOURCE_TABLE	VARCHAR(128)		
SOURCE_VIEW_QUAL	SMALLINT		
TARGET_SERVER	CHAR(18) NOT NULL		
TARGET_ALIAS	CHAR(8)		
TARGET_OWNER	VARCHAR(30) NOT NULL		
TARGET_TABLE	VARCHAR(128) NOT NULL		
CAPTURE_SCHEMA	VARCHAR(30) NOT NULL		
TGT_CAPTURE_SCHEMA	VARCHAR(30)		
FEDERATED_SRC_SRVR	VARCHAR(18)		
FEDERATED_TGT_SRVR	VARCHAR(18)		
JRN_LIB	CHAR(10)		
JRN_NAME	CHAR(10)		
COMMIT_COUNT	SMALLINT		
OPTION_FLAGS	CHAR(4) NOT NULL		
EVENT_NAME	CHAR(18)		
ENDTIME	TIMESTAMP NOT NULL WITH DEFAULT		
SOURCE_CONN_TIME	TIMESTAMP		
SQLSTATE	CHAR(5)		
SQLCODE	INT		
SQLERRP	CHAR(8)		
SQLERRM	VARCHAR(70)		
APPERRM	VARCHAR(760)		
		<b>ASN.IBMSNAP_APPLY_JOB</b> <i>OS/400 uniquement</i>	
		APPLY_QUAL	CHAR(18) NOT NULL
		CONTROL_SERVER	CHAR(18) NOT NULL
		JOB_NAME	CHAR(10) NOT NULL
		USER_NAME	CHAR(10) NOT NULL
		JOB_NUMBER	CHAR(6) NOT NULL
		<b>ASN.IBMSNAP_APPLYTRACE</b> (APPLY_QUAL, TRACE_TIME)	
		APPLY_QUAL	CHAR(18) NOT NULL
		TRACE_TIME	TIMESTAMP NOT NULL
		OPERATION	CHAR(8) NOT NULL
		DESCRIPTION	VARCHAR(1024) NOT NULL

Figure 10. Tables utilisées dans le serveur de contrôle Apply. Ces tables sont utilisées par le programme Apply dans le serveur de contrôle Apply. Les colonnes constituant l'index principal de chaque table sont affichées entre parenthèses sous le nom de la table.

## Tables de contrôle utilisées dans le serveur de contrôle Apply (image 2 de 2)

<b>ASN.IBMSNAP_SUBS_COLS</b> (APPLY_QUAL, SET_NAME, WHOS_ON_FIRST, TARGET_OWNER, TARGET_TABLE, TARGET_NAME)		<b>ASN.IBMSNAP_SUBS_STMTS</b> (APPLY_QUAL, SET_NAME, WHOS_ON_FIRST, BEFORE_OR_AFTER, STMT_NUMBER)	
APPLY_QUAL	CHAR(18) NOT NULL	APPLY_QUAL	CHAR(18) NOT NULL
SET_NAME	CHAR(18) NOT NULL	SET_NAME	CHAR(18) NOT NULL
WHOS_ON_FIRST	CHAR(1) NOT NULL	WHOS_ON_FIRST	CHAR(1) NOT NULL
TARGET_OWNER	VARCHAR(30) NOT NULL	BEFORE_OR_AFTER	CHAR(1) NOT NULL
TARGET_TABLE	VARCHAR(128) NOT NULL	STMT_NUMBER	SMALLINT NOT NULL
COL_TYPE	CHAR(1) NOT NULL	EI_OR_CALL	CHAR(1) NOT NULL
TARGET_NAME	VARCHAR(30) NOT NULL	SQL_STMT	VARCHAR(1024)
IS_KEY	CHAR(1) NOT NULL	ACCEPT_SQLSTATES	VARCHAR(50)
COLNO	SMALLINT NOT NULL		
EXPRESSION	VARCHAR(254) NOT NULL		
<b>ASN.IBMSNAP_SUBS_EVENT</b> (EVENT_NAME, EVENT_TIME)		<b>ASN.IBMSNAP_SUBS_MEMBR</b> (APPLY_QUAL, SET_NAME, WHOS_ON_FIRST, SOURCE_OWNER, SOURCE_TABLE, SOURCE_VIEW_QUAL, TARGET_OWNER, TARGET_TABLE)	
EVENT_NAME	CHAR(18) NOT NULL	APPLY_QUAL	CHAR(18) NOT NULL
EVENT_TIME	TIMESTAMP NOT NULL	SET_NAME	CHAR(18) NOT NULL
END_SYNCPOINT	CHAR(10) FOR BIT DATA	WHOS_ON_FIRST	CHAR(1) NOT NULL
END_OF_PERIOD	TIMESTAMP	SOURCE_OWNER	VARCHAR(30) NOT NULL
		SOURCE_TABLE	VARCHAR(128) NOT NULL
		SOURCE_VIEW_QUAL	SMALLINT NOT NULL
		TARGET_OWNER	VARCHAR(30) NOT NULL
		TARGET_TABLE	VARCHAR(128) NOT NULL
		TARGET_CONDENSED	CHAR(1) NOT NULL
		TARGET_COMPLETE	CHAR(1) NOT NULL
		TARGET_STRUCTURE	SMALLINT NOT NULL
		PREDICATES	VARCHAR(1024)
		MEMBER_STATE	CHAR(1)
		TARGET_KEY_CHG	CHAR(1) NOT NULL
		UOW_CD_PREDICATES	VARCHAR(1024)
		JOIN_UOW_CD	CHAR(1)
		LOADX_TYPE	SMALLINT
		LOADX_SRC_N_OWNER	VARCHAR(30)
		LOADX_SRC_N_TABLE	VARCHAR(128)
<b>ASN.IBMSNAP_SUBS_SET</b> (APPLY_QUAL, SET_NAME, WHOS_ON_FIRST)			
APPLY_QUAL	CHAR(18) NOT NULL		
SET_NAME	CHAR(18) NOT NULL		
SET_TYPE	CHAR(1) NOT NULL		
WHOS_ON_FIRST	CHAR(1) NOT NULL		
ACTIVATE	SMALLINT NOT NULL		
SOURCE_SERVER	CHAR(18) NOT NULL		
SOURCE_ALIAS	CHAR(8)		
TARGET_SERVER	CHAR(18) NOT NULL		
TARGET_ALIAS	CHAR(8)		
STATUS	SMALLINT NOT NULL		
LASTRUN	TIMESTAMP NOT NULL		
REFRESH_TYPE	CHAR(1) NOT NULL		
SLEEP_MINUTES	INT		
EVENT_NAME	CHAR(18)		
LASTSUCCESS	TIMESTAMP		
SYNCHPOINT	CHAR(10) FOR BIT DATA		
SYNCHTIME	TIMESTAMP		
CAPTURE_SCHEMA	VARCHAR(30) NOT NULL		
TGT_CAPTURE_SCHEMA	VARCHAR(30)		
FEDERATED_SRC_SRVR	VARCHAR(18)		
FEDERATED_TGT_SRVR	VARCHAR(18)		
JRN_LIB	CHAR(10)		
JRN_NAME	CHAR(10)		
OPTION_FLAGS	CHAR(4) NOT NULL		
COMMIT_COUNT	SMALLINT		
MAX_SYNCH_MINUTES	SMALLINT		
AUX_STMTS	SMALLINT NOT NULL		
ARCH_LEVEL	CHAR(4) NOT NULL		

Figure 11. Tables utilisées dans le serveur de contrôle Apply (suite). Ces tables sont utilisées par le programme Apply dans le serveur de contrôle Apply. Les colonnes constituant l'index principal de chaque table sont affichées entre parenthèses sous le nom de la table.



## Tables de contrôle utilisées dans le serveur de contrôle Monitor

<b>ASNJBMSNAP_ALERTS</b> (MONITOR_QUAL, COMPONENT, SERVER_NAME, SCHEMA_OR_QUAL, CONDITION_NAME, ALERT_CODE)		<b>ASNJBMSNAP_GROUPS</b> (GROUP_NAME)	
MONITOR_QUAL	CHAR(18) NOT NULL	GROUP_NAME	VARCHAR(127) NOT NULL
ALERT_TIME	TIMESTAMP NOT NULL	DESCRIPTION	VARCHAR(1024)
COMPONENT	CHAR(1) NOT NULL		
SERVER_NAME	CHAR(18) NOT NULL		
SERVER_ALIAS	CHAR(8)		
SCHEMA_OR_QUAL	VARCHAR(30) NOT NULL		
SET_NAME	CHAR(18) NOT NULL WITH DEFAULT		
CONDITION_NAME	CHAR(18) NOT NULL		
OCCURRED_TIME	TIMESTAMP NOT NULL		
ALERT_COUNTER	SMALLINT NOT NULL		
ALERT_CODE	CHAR(10) NOT NULL		
RETURN_CODE	INT NOT NULL		
NOTIFICATION_SENT	CHAR(1) NOT NULL		
ALERT_MESSAGE	VARCHAR(1024) NOT NULL		
<b>ASNJBMSNAP_CONDITIONS</b> (MONITOR_QUAL, SERVER_NAME, COMPONENT, SCHEMA_OR_QUAL, SET_NAME, CONDITION_NAME)		<b>ASNJBMSNAP_MONENQ</b> (MONITOR_QUAL)	
SERVER_NAME	CHAR(18) NOT NULL	MONITOR_QUAL	CHAR(18) NOT NULL
COMPONENT	CHAR(1) NOT NULL		
SCHEMA_OR_QUAL	VARCHAR(30) NOT NULL		
SET_NAME	CHAR(18) NOT NULL WITH DEFAULT		
MONITOR_QUAL	CHAR(18) NOT NULL		
SERVER_ALIAS	CHAR(8)		
ENABLED	CHAR(1) NOT NULL		
CONDITION_NAME	CHAR(18) NOT NULL		
PARAM_INT	INT		
PARAM_CHAR	VARCHAR(128)		
CONTACT_TYPE	CHAR(1) NOT NULL		
CONTACT	VARCHAR(127) NOT NULL		
<b>ASNJBMSNAP_CONTACTGRP</b> (GROUP_NAME, CONTACT_NAME)		<b>ASNJBMSNAP_MONSERVERS</b> (MONITOR_QUAL, SERVER_NAME)	
GROUP_NAME	VARCHAR(127) NOT NULL	MONITOR_QUAL	CHAR(18) NOT NULL
CONTACT_NAME	VARCHAR(127) NOT NULL	SERVER_NAME	CHAR(18) NOT NULL
		SERVER_ALIAS	CHAR(8)
		LAST_MONITOR_TIME	TIMESTAMP NOT NULL
		START_MONITOR_TIME	TIMESTAMP
		END_MONITOR_TIME	TIMESTAMP
		LASTRUN	TIMESTAMP NOT NULL
		LASTSUCCESS	TIMESTAMP
		STATUS	SMALLINT NOT NULL
<b>ASNJBMSNAP_CONTACTS</b> (CONTACT_NAME)		<b>ASNJBMSNAP_MONTRACE</b> (MONITOR_QUAL, TRACE_TIME)	
CONTACT_NAME	VARCHAR(127) NOT NULL	MONITOR_QUAL	CHAR(18) NOT NULL
EMAIL_ADDRESS	VARCHAR(128) NOT NULL	TRACE_TIME	TIMESTAMP NOT NULL
ADDRESS_TYPE	CHAR(1) NOT NULL	OPERATION	CHAR(8) NOT NULL
DELEGATE	VARCHAR(127)	DESCRIPTION	VARCHAR(1024) NOT NULL
DELEGATE_START	DATE		
DELEGATE_END	DATE		
DESCRIPTION	VARCHAR(1024)		
		<b>ASNJBMSNAP_MONTRAIL</b> (pas d'index unique)	
		MONITOR_QUAL	CHAR(18) NOT NULL
		SERVER_NAME	CHAR(18) NOT NULL
		SERVER_ALIAS	CHAR(8)
		STATUS	SMALLINT NOT NULL
		LASTRUN	TIMESTAMP NOT NULL
		LASTSUCCESS	TIMESTAMP
		ENDTIME	TIMESTAMP NOT NULL WITH DEFAULT
		LAST_MONITOR_TIME	TIMESTAMP NOT NULL
		START_MONITOR_TIME	TIMESTAMP
		END_MONITOR_TIME	TIMESTAMP
		SQLCODE	INT
		SQLSTATE	CHAR(5)
		NUM_ALERTS	INT NOT NULL
		NUM_NOTIFICATIONS	INT NOT NULL

Figure 12. Tables utilisées dans le serveur de contrôle Monitor. Ces tables sont utilisées par le programme Replication Alert Monitor dans le serveur de contrôle Monitor. Les colonnes constituant l'index principal de chaque table sont affichées entre parenthèses sous le nom de la table.

## Tables de contrôle utilisées dans le serveur de contrôle Monitor (image 2 de 2)

### ASN.IBMSNAP\_MONSERVERS

(MONITOR\_QUAL, SERVER\_NAME)

MONITOR_QUAL	CHAR(18) NOT NULL
SERVER_NAME	CHAR(18) NOT NULL
SERVER_ALIAS	CHAR(8)
LAST_MONITOR_TIME	TIMESTAMP NOT NULL
START_MONITOR_TIME	TIMESTAMP
END_MONITOR_TIME	TIMESTAMP
LASTRUN	TIMESTAMP NOT NULL
LASTSUCCESS	TIMESTAMP
STATUS	SMALLINT NOT NULL

### ASN.IBMSNAP\_MONTRACE

(MONITOR\_QUAL, TRACE\_TIME)

MONITOR_QUAL	CHAR(18) NOT NULL
TRACE_TIME	TIMESTAMP NOT NULL
OPERATION	CHAR(8) NOT NULL
DESCRIPTION	VARCHAR(1024) NOT NULL

### ASN.IBMSNAP\_MONTRAIL

(pas d'index)

MONITOR_QUAL	CHAR(18) NOT NULL
SERVER_NAME	CHAR(18) NOT NULL
SERVER_ALIAS	CHAR(8)
STATUS	SMALLINT NOT NULL
LASTRUN	TIMESTAMP NOT NULL
LASTSUCCESS	TIMESTAMP
ENDTIME	TIMESTAMP NOT NULL
	WITH DEFAULT
LAST_MONITOR_TIME	TIMESTAMP NOT NULL
START_MONITOR_TIME	TIMESTAMP
END_MONITOR_TIME	TIMESTAMP
SQLCODE	INT
SQLSTATE	CHAR(5)
NUM_ALERTS	INT NOT NULL
NUM_NOTIFICATIONS	INT NOT NULL

Figure 13. Tables utilisées dans le serveur de contrôle Monitor (suite). Ces tables sont utilisées par le programme Replication Alert Monitor dans le serveur de contrôle Monitor. Les colonnes constituant l'index principal de chaque table sont affichées entre parenthèses sous le nom de la table.

## Tables sur le serveur de contrôle de Capture

Les tables enregistrées sur le serveur de contrôle de Capture contiennent des informations sur vos sources enregistrées et sur la manière dont le programme Capture ou les déclencheurs traitent les sources.

Pour Linux, UNIX, Windows et z/OS, vous générez ces tables de contrôle en fonction de vos spécifications via le programme de ligne de commande ASNCLP ou le Centre de réplication. Pour System i, ces tables de contrôle sont créées automatiquement pour vous dans la bibliothèque ASN lorsque vous installez DataPropagator pour System i. Vous pouvez utiliser les commandes System i pour créer des tables de contrôle de Capture dans des schémas de capture de remplacement.

Le tableau 57 décrit les tables de contrôle sur le serveur Capture.

Tableau 57. Guide de référence pour les tables utilisées sur le serveur de contrôle de Capture

Nom de la table	Description
«Table IBMSNAP_CAPSCHEMAS», à la page 424	Contient les noms de tous les schémas de Capture
Table IBMSNAP_AUTHTKN (System i)	Contient des informations afin de prendre en charge la réplication bidirectionnelle.

Tableau 57. Guide de référence pour les tables utilisées sur le serveur de contrôle de Capture (suite)

Nom de la table	Description
<p><b>z/OS</b></p> <p><b>Linux UNIX Windows</b> «Table IBMSNAP_CAPENQ (z/OS, Linux, UNIX, Windows)», à la page 417</p>	<p>Pour chaque schéma de Capture, cette table est utilisée pour s'assurer que :</p> <ul style="list-style-type: none"> <li>• <b>Linux UNIX Windows</b> Pour DB2 pour Linux, UNIX et Windows, un seul programme Capture est en cours d'exécution par base de données.</li> <li>• <b>z/OS</b> Pour le non partage de données de DB2 pour z/OS, un seul programme Capture est en cours d'exécution par sous-système.</li> <li>• <b>z/OS</b> Pour le partage de données de DB2 pour z/OS, un seul programme Capture est en cours d'exécution par groupe de partage de données.</li> </ul>
«Table CD», à la page 428	Contient des informations sur les modifications qui ont été effectuées sur la source. Cette table n'est pas créée tant que vous n'avez pas enregistré une source de réplication.
«Table de modification des données (non DB2)», à la page 427	Contient des informations sur les modifications qui ont été effectuées sur la source et des colonnes supplémentaires pour identifier la classification séquentielle de ces modifications.
«Table IBMSNAP_CAPMON», à la page 418	Contient des statistiques opérationnelles qui permettent de surveiller le programme Capture.
«Table IBMSNAP_CAPPARMS», à la page 420	Contient les paramètres que vous avez définis pour contrôler les opérations d'un programme Capture.
«Table IBMSNAP_CAPTRACE», à la page 426	Contient des messages provenant du programme Capture.
«Table IBMQREP_IGNTRAN», à la page 429	Peut s'utiliser pour aviser le programme Capture des transactions que vous ne souhaitez pas capturer depuis le journal de récupération de DB2.
«Table IBMQREP_IGNTRANTRC», à la page 430	Enregistre des informations sur les transactions qui ont été spécifiées à ignorer.
Table IBMQREP_PART_HIST	Conserve un historique des modifications apportées aux tables source partitionnées sur les systèmes Linux, UNIX et Windows. Cette table est utilisée par la réplication SQL et la réplication Q.
«Table IBMSNAP_PARTITIONINFO», à la page 431	Contient des informations qui permettent au programme Capture de redémarrer à partir du premier numéro d'ordre du journal nécessaire.
«Table IBMSNAP_PRUNE_LOCK», à la page 434	Utilisé pour sérialiser l'accès du programme Capture des tables de modification des données lors d'un démarrage à froid ou lors d'un élagage en fonction de la durée de conservation (élagage lorsque la limite de rétention est atteinte ou dépassée).
«Table IBMSNAP_PRUNE_SET», à la page 435	Coordonne l'élagage des tables de modification des données.
«Table IBMSNAP_PRUNCNTL», à la page 432	Coordonne les mises à jour des points de synchronisation entre les programmes Capture et Apply.

Tableau 57. Guide de référence pour les tables utilisées sur le serveur de contrôle de Capture (suite)

Nom de la table	Description
<b>System i</b> IBMSNAP_REG_EXT (System i)	Une extension de la table de registres. Contient des informations supplémentaires sur les sources de réplication, telles que le nom de journal et le nom d'entrée de base de données de la table source éloignée.
«Table IBMSNAP_REGISTER», à la page 437	Contient des informations sur les sources de réplication, telles que les noms des tables sources de réplication, leurs attributs et les tables de modification des données et les noms de table CCD correspondants.
«Table IBMSNAP_REG_SYNCH (relationnelle non DB2)», à la page 444	Utilisé lors de la réplication à partir d'une source de données relationnelles non-DB2. Un déclencheur UPDATE sur cette table simule le programme Capture en lançant une mise à jour de la valeur SYNCHPOINT pour toutes les lignes de la table de registres avant que le programme Apply ne lise les informations à partir de la table de registres.
«Table IBMSNAP_RESTART», à la page 444	Contient des informations qui permettent au programme Capture de reprendre à partir du point correct dans le journal. Pour les environnements System i, cette table est également utilisée pour déterminer l'heure de début de la commande <b>RCVJRNE</b> (Receive Journal Entry).
«Table IBMSNAP_SEQTABLE (Informix)», à la page 447	Contient une séquence de nombres uniques que la réplication SQL utilise comme équivalent des numéros d'ordre du journal pour les tables Informix.
«Table IBMSNAP_SIGNAL», à la page 447	Contient tous les signaux utilisés pour demander le programme Capture. Ces signaux peuvent être envoyés manuellement ou par le programme Apply.
«Table IBMSNAP_UOW», à la page 451	Fournit des informations supplémentaires sur les transactions qui ont été validées sur une table source.

## Table IBMSNAP\_AUTHTKN (System i)

La table IBMSNAP\_AUTHTKN est utilisée uniquement dans l'environnement System i. Cette table est utilisée pendant la réplication update-anywhere afin de conserver une trace des transactions qui ont été traitées par un programme Apply particulier. Le programme Capture élague cette table en fonction de la durée de conservation que vous avez définie.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : JRN\_LIB, JRN\_NAME

**Important** : prenez garde au moment de mettre à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 58, à la page 417 fournit une brève description des colonnes de la table IBMSNAP\_AUTHTKN.

Tableau 58. Colonnes dans la table IBMSNAP\_AUTHTKN

Nom de colonne	Description
APPLY_QUAL	<p><b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non</p> <p>Le qualificatif Apply qui identifie le programme Apply qui a traité la transaction. Ce qualificatif est utilisé pendant la réplication update-anywhere pour éviter au programme Apply de répliquer les mêmes modifications à l'infini.</p>
IBMSNAP_AUTHTKN	<p><b>Type de données</b> : CHAR(26) ; <b>Null admis</b> : non</p> <p>Le nom de travail associé à la transaction. Capture pour System i associe le nom de cette colonne au nom de travail qui a émis la transaction afin de déterminer si la transaction a été émise par le programme Apply ou une application utilisateur. Si les noms de travail correspondent, alors, Capture pour System i copie le qualificatif Apply qui se trouve dans la colonne APPLY_QUAL de cette table dans la colonne APPLY_QUAL dans la ligne correspondante de la table des unités d'oeuvre. Si les noms ne correspondent pas, alors Capture pour System i définit la colonne APPLY_QUAL de la table des unités d'oeuvre sur null. Cette colonne n'est pas copiée automatiquement dans les autres tables ; vous devez la sélectionner et la copier en tant que colonne de données utilisateur.</p>
JRN_LIB	<p><b>Type de données</b> : CHAR(10) ; <b>Null admis</b> : non</p> <p>Nom de bibliothèque du journal d'où provenait les transactions.</p>
JRN_NAME	<p><b>Type de données</b> : CHAR(10) ; <b>Null admis</b> : non</p> <p>Nom du journal d'où provenaient les transactions.</p>
IBMSNAP_LOGMARKER	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non</p> <p>L'heure approximative à laquelle la transaction a été validée sur le serveur de contrôle Capture.</p>

## Table IBMSNAP\_CAPENQ (z/OS, Linux, UNIX, Windows)

Pour un schéma de Capture simple, la table IBMSNAP\_CAPENQ permet de s'assurer qu'un seul programme Capture est exécuté par base de données, sous-système ou groupe de partage de données.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : aucun

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

La table IBMSNAP\_CAPENQ n'est pas utilisée sur des serveurs relationnels non-DB2 ou System i.

En cours d'exécution, le programme Capture verrouille exclusivement cette table.

Le tableau 59, à la page 418 fournit une brève description de la colonne dans la table IBMSNAP\_CAPENQ.

Tableau 59. Colonne dans la table IBMSNAP\_CAPENQ

Nom de colonne	Description
LOCKNAME	<b>Type de données</b> : CHAR(9) ; <b>Null admis</b> : non  Cette colonne ne contient aucune donnée.

## Table IBMSNAP\_CAPMON

Le programme Capture insère une ligne dans la table IBMSNAP\_CAPMON après chaque intervalle afin de vous fournir des statistiques opérationnelles. Le Centre de réplication utilise des informations dans cette table (et dans d'autres tables) afin que vous puissiez surveiller l'état du programme Capture.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : MONITOR\_TIME

Dans la table IBMSNAP\_CAPPARMS, la valeur que vous spécifiez pour MONITOR\_INTERVAL indique la fréquence à laquelle le programme Capture fait des insertions dans la table de contrôle Capture et la valeur que vous spécifiez pour MONITOR\_LIMIT indique le nombre de minutes pendant lesquelles les lignes restent dans la table avant d'être susceptibles d'être élaguées.

Le tableau 60 fournit une brève description des colonnes dans la table IBMSNAP\_CAPMON.

Tableau 60. Colonnes dans la table IBMSNAP\_CAPMON

Nom de colonne	Description
MONITOR_TIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non  Horodatage (sur le serveur Capture) lorsque la ligne a été insérée dans cette table.
RESTART_TIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non  Horodatage lorsque l'appel actuel du programme Capture a été redémarré.
CURRENT_MEMORY	<b>Type de données</b> : INT ; <b>Null admis</b> : non  La quantité de mémoire (en octets) que le programme Capture a utilisée.
CD_ROWS_INSERTED	<b>Type de données</b> : INT ; <b>Null admis</b> : non  Le nombre de lignes que le programme Capture a inséré dans la table de modification des données pour toutes les tables sources.
RECAP_ROWS_SKIPPED	<b>Type de données</b> : INT ; <b>Null admis</b> : non  Pour la réplication bidirectionnelle, il s'agit du nombre de lignes que le programme Capture a traité mais n'a pas inséré dans la table de modification des données. Les lignes ont été ignorées car l'enregistrement a été défini afin que le programme Capture ne réenregistre pas les modifications qui ont été répliquées sur cette table ne provenant pas de ce serveur source.

Tableau 60. Colonnes dans la table IBMSNAP\_CAPMON (suite)

Nom de colonne	Description
TRIGR_ROWS_SKIPPED	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre de lignes que le programme Capture a traité mais n'a pas inséré dans la table de modification des données. Les lignes ont été ignorées car vous avez défini un déclencheur sur l'enregistrement afin que le programme Capture supprime certaines lignes.</p>
CHG_ROWS_SKIPPED	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre de lignes que le programme Capture a traité mais n'a pas inséré dans la table de modification des données. Les lignes ont été ignorées car l'enregistrement a été défini afin que le programme Capture ne capture que les modifications effectuées dans des colonnes enregistrées.</p>
TRANS_PROCESSED	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre de transactions sur le système source que le programme Capture a traitées.</p>
TRANS_SPILLED	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre de transactions sur le système source que le programme Capture a déversées sur le disque en raison de restrictions de mémoire.</p>
MAX_TRAN_SIZE	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>La plus grande transaction qui s'est produite sur le système source. La connaissance de la taille de la transaction peut vous inciter à modifier les paramètres mémoire.</p>
LOCKING_RETRIES	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre de fois où un interblocage a entraîné une transformation.</p>
JRN_LIB (System i)	<p><b>Type de données</b> : CHAR(10) ; <b>Null admis</b> : oui</p> <p><b>System i</b> Nom de bibliothèque du journal que le programme Capture traitait.</p>
JRN_NAME (System i)	<p><b>Type de données</b> : CHAR(10) ; <b>Null admis</b> : oui</p> <p>Nom du journal que le programme Capture traitait.</p>
LOGREADLIMIT	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre de fois où le programme Capture a fait une pause dans la lecture des enregistrements de journal car 1000 enregistrements ont été lus mais aucune transaction terminée n'a encore été rencontrée dans ces 1000 enregistrements.</p>
CAPTURE_IDLE	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre de fois où le programme Capture s'est mis en veille car il n'avait aucun travail à traiter.</p>
SYNCHTIME	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non</p> <p>Valeur actuelle de SYNCHTIME lue à partir de la ligne globale de la table de registres lorsque l'enregistrement de contrôle a été inséré dans cette table.</p>
CURRENT_LOG_TIME	<p><b>Type de données</b> : TIMESTAMP; <b>Null admis</b>: Non</p> <p>Horodatage sur le serveur Capture de la dernière validation de la base de données vue par le programme de lecture de journal Capture.</p>

Tableau 60. Colonnes dans la table IBMSNAP\_CAPMON (suite)

Nom de colonne	Description
RESTART_SEQ	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA; <b>NULL admis</b> : non, avec valeur par défaut</p> <p>Numéro LSN logique dans le journal de récupération auquel le programme Capture démarre pendant un redémarrage à chaud. Cette valeur représente le premier numéro d'ordre du journal que le programme Capture a trouvé pour lequel aucun enregistrement de validation ou d'abandon n'a encore été trouvé.</p>
CURRENT_SEQ	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA; <b>NULL admis</b> : non, avec valeur par défaut</p> <p>Numéro LSN logique le plus récent dans le journal de récupération lu par le programme Capture.</p>
LAST_EOL_TIME	<p><b>Type de données</b> : TIMESTAMP ; <b>NULL admis</b> : oui</p> <p>Heure sur le serveur Capture à laquelle il a atteint la fin du journal.</p>
LOGREAD_API_TIME	<p><b>Type de données</b> : INTEGER; <b>Null admis</b> : oui</p> <p>Nombre de millisecondes que le programme Capture a passé à utiliser l'API de lecture du journal DB2 pour extraire les enregistrements de journaux.</p>
NUM_LOGREAD_CALLS	<p><b>Type de données</b> : INTEGER; <b>Null admis</b> : oui</p> <p>Nombre d'appels d'API de lecture de journal effectués par le programme Capture.</p>

## Table IBMSNAP\_CAPPARMS

La table IBMSNAP\_CAPPARMS contient des paramètres que vous pouvez modifier pour contrôler les opérations du programme Capture. Vous pouvez définir ces paramètres pour définir des valeurs telles que la longueur de temps pendant laquelle le programme Capture garde des données dans les tables de modification des données et les tables UOW avant élagage et la quantité de temps pendant laquelle le programme Capture est autorisé à décaler le traitement des enregistrements de journal. Si vous modifiez ces paramètres dans cette table, le programme Capture ne lira vos modifications que lors du démarrage.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : aucun

Cette table contient des informations que vous pouvez mettre à jour à l'aide de SQL.

Le tableau 61, à la page 421 fournit une brève description des colonnes dans la table IBMSNAP\_CAPPARMS.



Tableau 61. Colonnes dans la table IBMSNAP\_CAPPARMS

Nom de colonne	Description
RETENTION_LIMIT	<p><b>Type de données</b> : INT ; <b>Null admis</b> : oui</p> <p>La longueur de temps pendant laquelle les lignes restent dans les tables de modification des données, les tables UOW et les tables de signaux avant qu'elles ne soient susceptibles d'être élaguées, dans les cas où elles n'ont pas été élaguées suivant des critères normaux. Normalement, les lignes de modification des données et les lignes UOW sont élaguées après avoir été appliquées à toutes les cibles et les lignes de signaux sont élaguées lorsque leur cycle est terminé (SIGNAL_STATE = C).</p>
LAG_LIMIT	<p><b>Type de données</b> : INT ; <b>Null admis</b> : oui</p> <p>Nombre de minutes pendant lesquelles le programme Capture est autorisé à décaler le traitement des enregistrements de journal avant de se fermer. Pendant les périodes où la fréquence de mise à jour est élevée, des régénérations intégrales peuvent être plus économiques que des mises à jour.</p>
COMMIT_INTERVAL	<p><b>Type de données</b> : INT ; <b>Null admis</b> : oui</p> <p>Fréquence, en secondes, à laquelle le programme Capture valide des données dans les tables de contrôle Capture, y compris les tables UOW et les tables de modification des données. Cette valeur doit être inférieure à la valeur de verrouillage DB2 afin d'éviter des conflits entre Capture et les unités d'exécution d'élagage.</p>
PRUNE_INTERVAL	<p><b>Type de données</b> : INT ; <b>Null admis</b> : oui</p> <p>Fréquence, en secondes, à laquelle le programme Capture élague automatiquement (AUTOPRUNE = Y) des lignes dans les tables de modification des données, les tables UOW, les tables de signaux, les tables de traces et les tables de contrôle de Capture qui ne sont plus nécessaires. Un intervalle d'élagage inférieur permet d'économiser de l'espace, mais augmente les coûts de traitement. Un intervalle d'élagage supérieur nécessite davantage d'espace de tables de modification des données et de tables UOW, mais diminue les coûts de traitement.</p>
TRACE_LIMIT	<p><b>Type de données</b> : INT ; <b>Null admis</b> : oui</p> <p>Nombre de minutes pendant lesquelles les lignes restent dans la table IBMSNAP_CAPTRACE avant d'être susceptibles d'être élaguées. Lors du processus d'élagage, les lignes dans la table de trace Capture sont élaguées si le nombre de minutes (horodatage actuel - heure à laquelle une ligne a été insérée dans la table de trace Capture) dépasse la valeur de TRACE_LIMIT.</p>
MONITOR_LIMIT	<p><b>Type de données</b> : INT ; <b>Null admis</b> : oui</p> <p>Nombre de minutes pendant lesquelles les lignes restent dans la table IBMSNAP_CAPMON avant d'être susceptibles d'être élaguées. Lors du processus d'élagage, des lignes dans la table de contrôle Capture sont élaguées si le nombre de minutes (horodatage actuel - MONITOR TIME) dépasse la valeur de MONITOR_LIMIT.</p>
MONITOR_INTERVAL	<p><b>Type de données</b> : INT ; <b>Null admis</b> : oui</p> <p>Fréquence, en secondes, à laquelle l'unité d'exécution de contrôle ajoute une ligne à la table IBMSNAP_CAPMON de contrôle Capture. Pour Capture pour System i, saisissez un intervalle supérieur à 120 secondes.</p>
MEMORY_LIMIT	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : oui</p> <p>Quantité de mémoire, en mégaoctets, que le programme Capture est autorisé à utiliser. Après l'utilisation de cette attribution, les transactions en mémoire se déverseront dans un fichier.</p>

Tableau 61. Colonnes dans la table IBMSNAP\_CAPPARMS (suite)

Nom de colonne	Description
REMOTE_SRC_SERVER	<p><b>Type de données</b> : CHAR(18) ; <b>Valeur NULL admise</b> : oui</p> <p>Réservé aux options ultérieures de la réplication SQL. Actuellement, cette colonne contient la valeur par défaut de null.</p>
AUTOPRUNE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si le programme Capture élague automatiquement des lignes qui ne sont plus nécessaires des tables de modification des données, des tables UOW, des tables de signaux, des tables de traces et des tables de contrôle de Capture :</p> <p>Y L'élagage automatique est activé.</p> <p>N L'élagage automatique n'est pas activé.</p>
TERM	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si le programme Capture se termine ou est mis en repos en même temps que DB2 :</p> <p>Y Le programme Capture se termine en cas d'arrêt ou de mise au repos de DB2.</p> <p>N Le programme Capture reste actif et attend le redémarrage de DB2.</p>
AUTOSTOP	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si le programme Capture arrête de capturer des modifications dès qu'il atteint la fin des journaux actifs :</p> <p>Y Le programme Capture s'arrête dès qu'il atteint la fin des journaux actifs.</p> <p>N Le programme Capture continue son exécution quand il atteint la fin des journaux actifs.</p>
LOGREUSE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si le programme Capture écrase le fichier journal Capture ou fait des ajouts à ce journal.</p> <p>Y Le programme Capture réutilise le fichier journal en le supprimant d'abord puis en le recréant lorsque le programme Capture est redémarré.</p> <p>N Le programme Capture ajoute de nouvelles informations au fichier journal Capture.</p>
LOGSTDOUT	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant où le programme Capture dirige les messages du fichier journal :</p> <p>Y Le programme Capture dirige les messages du fichier journal vers la sortie standard (STDOUT) et vers le fichier journal.</p> <p>N Le programme Capture dirige la plupart des messages de fichier journal vers le fichier journal uniquement. Les messages d'initialisation vont à la fois à la sortie standard (STDOUT) et au fichier journal.</p>

Tableau 61. Colonnes dans la table IBMSNAP\_CAPPARMS (suite)

Nom de colonne	Description
<b>z/OS</b>	<b>Type de données</b> : SMALLINT ; <b>Null admis</b> : oui
<b>Linux UNIX Windows</b> SLEEP_INTERVAL (z/OS, Linux, UNIX, Windows)	Nombre de secondes pendant lesquelles le programme Capture est mis en veille lorsqu'il atteint la fin des journaux actifs (sous Linux, UNIX et Windows, ou dans des environnements sans partage de données z/OS), ou lorsqu'une quantité inefficace de données a été renvoyée (dans les environnements de partage des données z/OS).
CAPTURE_PATH	<b>Type de données</b> : VARCHAR(1040) ; <b>Valeur NULL admise</b> : Oui  Chemin où est envoyée la sortie du programme Capture.
STARTMODE	<b>Type de données</b> : VARCHAR(10) ; <b>Null admis</b> : oui  Procédure de traitement que le programme Capture utilise lorsqu'il est démarré :  <b>cold</b> Le programme Capture supprime toutes les lignes dans ses tables de modification des données et dans sa table UOW lors de l'initialisation. Tous les abonnements à ces sources de réplication sont intégralement régénérés lors du cycle de traitement Apply suivant (c'est-à-dire que toutes les données sont copiées des tables sources dans les tables cibles). Si le programme Capture tente un démarrage à froid alors que vous avez désactivé la régénération intégrale, le programme Capture redémarrera mais le programme Apply échouera et émettra un message d'erreur.  <b>warmsi</b> Le programme Capture démarre à chaud, sauf si c'est la première fois que vous démarrez le programme Capture ; dans ce cas, il passe à un démarrage à froid. Le mode de démarrage warmsi assure que le démarrage à froid ne se produit que lors du lancement initial du programme Capture.  <b>warmns</b> Le programme Capture démarre à chaud. S'il ne peut pas être démarré à chaud, il ne passe pas à un démarrage à froid. Le mode de démarrage warmns permet d'éviter que des démarrages à froid ne se produisent de façon inattendue et est utile lorsque des problèmes surviennent (tels que des bases de données ou des espaces tables indisponibles) nécessitant une réparation et empêchant un démarrage à chaud. Lorsque le programme Capture démarre à chaud, il reprend le traitement là où il s'était arrêté. Si des erreurs se produisent après le démarrage du programme Capture, le programme Capture s'arrête sans modifier aucune table.
ARCH_LEVEL	<b>Type de données</b> : CHAR(4) ; <b>Null admis</b> : non  La version des tables de contrôle Capture : <b>1001</b> Version 10 sous Linux, UNIX et Windows <b>100Z</b> Version 10 sous z/OS <b>0973</b> Version 9.7 avec groupe de correctifs 3 sous Linux, UNIX et Windows  <b>Avertissement</b> : Au moment de la mise à jour de la table IBMSNAP_CAPPARMS, ne modifiez pas la valeur dans cette colonne.

Tableau 61. Colonnes dans la table IBMSNAP\_CAPPARMS (suite)

Nom de colonne	Description
COMPATIBILITY	<p><b>Type de données</b> : CHAR(4); <b>Null admis</b> : non, valeur par défaut</p> <p>Détermine la longueur des numéros de séquence du journal dans les tables de contrôle Capture, les tables de modification des données et les tables UOW. Les numéros de séquence du journal de 16 octets sont utilisés à partir de la version 10 sous Linux, UNIX et Windows.</p> <p><b>1001</b> Les numéros de séquence du journal de 16 octets sont utilisés</p> <p><b>0801</b> Les numéros de séquence du journal de 10 octets sont utilisés</p> <p>Le programme Apply utilise la valeur dans cette colonne pour déterminer la longueur des numéros de séquence du journal à utiliser dans ses tables de contrôle et ses tables CCD cibles.</p>
LOGRDBUFSZ	<p><b>Type de données</b> : INTEGER ; <b>Null admis</b>: non, avec valeur par défaut</p> <p><b>Linux, UNIX et Windows uniquement</b> : la taille en ko du tampon que le programme Capture envoie à DB2 lorsque Capture récupère les enregistrements de journal. DB2 remplit la mémoire tampon avec les enregistrements de journaux disponibles que Capture n'a pas extrait. Valeur par défaut : 256 ko.</p>

## Table IBMSNAP\_CAPSCHEMAS

La table IBMSNAP\_CAPSCHEMAS conserve les noms de tous les schémas de Capture. Elle permet aux outils d'administration et aux autres utilitaires de trouver rapidement toutes les tables pour un serveur de contrôle de Capture donné. Une ligne est automatiquement insérée chaque fois que vous créez un nouveau schéma de Capture.

**Serveur** : serveur de contrôle Capture

**Index** : CAP\_SCHEMA\_NAME

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus.

Les deux tables suivantes montrent les agencements spécifiques aux systèmes d'exploitation de la table IBMSNAP\_CAPSCHEMAS.

Tableau 62. Colonnes dans la table IBMSNAP\_CAPSCHEMAS pour tous les systèmes d'exploitation autres que System i

Nom de colonne	Description
CAP_SCHEMA_NAME	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : Oui</p> <p>Nom d'un schéma de Capture. Une ligne existe pour chaque schéma de Capture.</p>

Tableau 63. Colonnes dans la table de schémas de Capture pour System i

Nom de colonne	Description
CAP_SCHEMA_NAME	<p><b>Type de données</b> : VARCHAR(30) ; <b>Null admis</b> : oui</p> <p>Nom d'un schéma de Capture. Une ligne existe pour chaque schéma de Capture.</p>

Tableau 63. Colonnes dans la table de schémas de Capture pour System i (suite)

Nom de colonne	Description
STATUS	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si le programme Capture identifié par ce schéma de Capture est en cours d'exécution :</p> <p>Y Le programme Capture est en cours d'exécution.</p> <p>N Le programme Capture n'est pas en cours d'exécution.</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 64 propose une brève description des colonnes de la table IBMQREP\_COLVERSION.

Tableau 64. Colonnes de la table IBMQREP\_COLVERSION

Nom de colonne	Description
LSN	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : 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.</p>
TABLEID1	<p><b>Type de données</b> : SMALLINT ; <b>NULL admis</b> : non</p> <p>Identificateur d'objet (OBID) dans SYSIBM.SYSTABLES.</p>
TABLEID2	<p><b>Type de données</b> : SMALLINT ; <b>NULL admis</b> : non</p> <p>Identificateur de base de données (DBID) dans SYSIBM.SYSTABLES.</p>
POSITION	<p><b>Type de données</b> : SMALLINT ; <b>NULL admis</b> : non</p> <p>Position ordinale de la colonne dans la table, 0 désignant la première colonne dans la table.</p>
NAME	<p><b>Type de données</b> : VARCHAR(128) ; <b>NULL admis</b> : non</p> <p>Nom de la colonne.</p>

Tableau 64. Colonnes de la table IBMQREP\_COLVERSION (suite)

Nom de colonne	Description
TYPE	<b>Type de données</b> : SMALLINT ; <b>NULL admis</b> : non  Identificateur de type de donnée interne pour la colonne (SQLTYPE dans SYSIBM.SYSCOLUMNS).
LENGTH	<b>Type de données</b> : INTEGER ; <b>NULL admis</b> : Non  Longueur maximale de donnée pour cette colonne.
NULLS	<b>Type de données</b> : CHAR(1); <b>Null admis</b> : 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	<b>Type de données</b> : VARCHAR(1536) ; <b>NULL admis</b> : 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	<b>Type de données</b> : INTEGER ; <b>NULL admis</b> : 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	<b>Type de données</b> : INTEGER ; <b>NULL admis</b> : oui  L'échelle des données décimales dans les colonnes décimales. La valeur est 0 pour les colonnes non décimales. Valeur par défaut : NULL

## Table IBMSNAP\_CAPTRACE

La table de trace Capture contient des messages provenant du programme Capture.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : TRACE\_TIME

Les deux tables suivantes montrent les agencements spécifiques aux systèmes d'exploitation de la table IBMSNAP\_CAPTRACE.

Linux UNIX Windows

z/OS

Tableau 65. Colonnes dans la table IBMSNAP\_CAPTRACE pour Linux, UNIX, Windows et z/OS

Nom de colonne	Description
OPERATION	<b>Type de données</b> : CHAR(8) ; <b>Null admis</b> : non  Le type d'opération de programme Capture, par exemple initialisation, capture ou erreur.
TRACE_TIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non  Heure sur le serveur de contrôle de Capture à laquelle la ligne a été insérée dans le table de trace Capture.

Tableau 65. Colonnes dans la table IBMSNAP\_CAPTRACE pour Linux, UNIX, Windows et z/OS (suite)

Nom de colonne	Description
DESCRIPTION	<p><b>Type de données</b> : VARCHAR(1024) ; <b>Null admis</b> : non</p> <p>L'identificateur de message suivi par le texte de message. Il peut s'agir d'un message d'erreur, d'un message d'avertissement ou d'un message d'information. Cette colonne contient un texte en anglais uniquement.</p>

**System i**

Tableau 66. Colonnes dans la table de trace de Capture pour System i

Nom de colonne	Description				
OPERATION	<p><b>Type de données</b> : CHAR(8) ; <b>Null admis</b> : non</p> <p>Type d'opération que le programme Capture a réalisée, par exemple, initialisation, capture ou cas d'erreur.</p>				
TRACE_TIME	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non</p> <p>Heure à laquelle la ligne a été insérée dans la table de trace Capture. Les lignes TRACE_TIME qui sont susceptibles d'être élaguées en fonction de la limite de trace sont effacées lorsque le programme Capture élague les tables de modification de données et les tables UOW.</p>				
JOB_NAME	<p><b>Type de données</b> : CHAR(26) ; <b>Null admis</b> : non</p> <p>Nom qualifié complet du travail qui a écrit cette entrée de trace.</p> <p><b>Position</b></p> <table border="0"> <thead> <tr> <th style="text-align: left;">Description</th> </tr> </thead> <tbody> <tr> <td>1–10    Nom de schéma de Capture ou nom de travail de consignation</td> </tr> <tr> <td>11–20    ID de l'utilisateur qui a démarré le programme Capture</td> </tr> <tr> <td>21–26    Numéro de travail</td> </tr> </tbody> </table>	Description	1–10    Nom de schéma de Capture ou nom de travail de consignation	11–20    ID de l'utilisateur qui a démarré le programme Capture	21–26    Numéro de travail
Description					
1–10    Nom de schéma de Capture ou nom de travail de consignation					
11–20    ID de l'utilisateur qui a démarré le programme Capture					
21–26    Numéro de travail					
JOB_STR_TIME	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non</p> <p>Heure de début du travail nommé dans le colonne JOB_NAME.</p>				
DESCRIPTION	<p><b>Type de données</b> : VARCHAR(298) ; <b>Null admis</b> : non</p> <p>L'identificateur de message suivi par le texte de message. L'identificateur de message est le premier des sept caractères de la colonne DESCRIPTION. Le texte de message commence à la neuvième position de la colonne DESCRIPTION.</p>				

## Table de modification des données (non DB2)

Les tables de modification cohérente des données (CCD) sur le serveur de contrôle de Capture sont des tables des informations sur les modifications qui ont été effectuées sur une source non DB2 et des colonnes supplémentaires pour identifier la classification séquentielle de ces modifications. Une table de modification cohérente des données sur le serveur de contrôle de Capture est une table remplie par un programme différent du programme Apply.

**Serveur** : serveur de contrôle Capture

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des pertes de données.

Le serveur de contrôle de Capture peut être :

- Une table de modification cohérente des données interne pour une source relationnelle non DB2.

Pour une réplication en mode capture des modifications, les déclencheurs de capture insèrent des modifications dans cette table quand des mises à jour surviennent sur la source relationnelle non DB2. Le nom de ce type de table de modification cohérente des données est stocké sur la même ligne dans la table IBMSNAP\_REGISTER quand la source de réplication qu'elle conserve est modifiée. Cette table est élaguée automatiquement par le déclencheur d'élagage créé lorsque vous enregistrez une source relationnelle non DB2.

- Table de modification cohérente des données externe pour les données non relationnelles et multiconstructeur.

Les programmes externes ne peuvent pas créer de tables de modification cohérente des données utilisables par la réplication SQL comme sources de réplication. Par exemple, IMS DataPropagator capture les modifications IMS dans une table de modification cohérente des données afin que les copies des données IMS puissent être recrées dans une base de données relationnelle. Les programmes externes doivent initialiser, gérer et fournir les valeurs correctes pour les colonnes de contrôle. Si vous possédez des tables de modification cohérente des données à remplissage externe qui ne sont pas gérées par un programme tel que IMS DataPropagator ou DataRefresher, vous devez gérer ces tables vous-même afin que le programme Apply puisse lire les tables de modification cohérente des données en tant que sources et fonctionne correctement.

Le tableau 67 fournit une brève description des colonnes dans la table de modification cohérente des données.

Tableau 67. Colonnes dans la table CCD

Nom de colonne	Description
IBMSNAP_INTENTSEQ	Un numéro de séquence qui identifie uniquement un changement. Cette valeur est croissante.
IBMSNAP_OPERATION	Indicateur signalant le type d'opération pour un enregistrement : <b>I</b> Insertion <b>U</b> Mise à jour <b>D</b> Suppression
IBMSNAP_COMMITSEQ	Numéro de séquence fournissant un ordre transactionnel.
IBMSNAP_LOGMARKER	L'heure approximative à laquelle les données ont été validées.
<i>colonnes clé utilisateur</i>	Si la table CCD est condensée, cette colonne contient les colonnes qui constituent la clé cible.
<i>colonnes utilisateur non clé</i>	Colonnes non clés de la table source. Les noms de colonne qui se trouvent dans la table source ne correspondent pas nécessairement à ces noms de colonne, mais les types de données doivent être compatibles.
<i>colonnes définies par l'utilisateur</i>	Colonnes définies par l'utilisateur, dérivées d'expressions SQL. Vous pouvez utiliser ce type de colonne avec des fonctions SQL pour convertir des types de données source en différents types de données cible.

## Table CD

Les tables de modification des données enregistrent toutes les modifications validées faites sur une source de réplication. L'élagage de la table de modification des données est coordonné par la table IBMSNAP\_PRUNE\_SET. Contrairement aux autres tables de contrôle Capture, les tables de modification des données sont



créées lorsque vous définissez une source de réplication ; elles ne sont pas créées automatiquement lorsque vous générez les tables de contrôle pour le serveur de contrôle de Capture.

**Serveur** : serveur de contrôle Capture

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des pertes de données.

Le tableau 68 fournit une liste et une brève description des colonnes dans la table de modification des données.

Tableau 68. Colonnes dans la table de modification des données

Nom de colonne	Description
IBMSNAP_COMMITSEQ	Numéro LSN de l'instruction de validation capturée. Cette colonne, qui se trouve également dans la table UOW, est incluse dans la table de modification des données pour permettre au programme Apply de traiter les tables cibles de copie utilisateur sans avoir besoin de joindre la table de modification des données à la table UOW. Lorsque la table de modification des données et la table UOW doivent être reliées, cette opération est effectuée par la colonne IBMSNAP_COMMITSEQ.
IBMSNAP_INTENTSEQ	Numéro d'ordre du journal de l'enregistrement de journal de la modification (insertion, mise à jour ou suppression). Cette valeur est globalement croissante. Si vous avez défini que les mises à jour doivent être traitées par paires suppression/insertion, la valeur IBMSNAP_INTENTSEQ pour la ligne de suppression est conçue de façon à être légèrement plus petite que la valeur correspondante pour la ligne d'insertion.
IBMSNAP_OPERATION	Indicateur signalant le type d'opération pour un enregistrement : <b>I</b> Insertion <b>U</b> Mise à jour <b>D</b> Suppression
<i>colonne utilisateur image après</i>	Dans la plupart des cas, la colonne image après contient la valeur qui est dans la colonne source après la modification. Cette colonne possède le même nom, le même type de données et les mêmes attributs null que la colonne source. Dans le cas d'une mise à jour, cette colonne reflète la nouvelle valeur des données qui ont été mises à jour. Dans le cas d'une suppression, cette colonne reflète la valeur des données qui ont été supprimées. Dans le cas d'une insertion, cette colonne reflète la nouvelle valeur des données qui ont été insérées.
<i>colonne utilisateur image avant</i>	Cette colonne existe uniquement dans la table de modification des données si vous avez enregistré la source de façon à ce qu'elle inclue des valeurs de colonnes image avant. Dans la plupart des cas, la colonne image avant contient la valeur qui était dans la colonne source avant la modification. Cette colonne possède le même nom que la colonne source, avec comme préfixe la valeur dans la colonne BEFORE_IMG_PREFIX dans la table IBMSNAP_REGISTER. Elle a également le même type de données que la colonne source ; cependant, elle admet toujours des valeurs null pour les opérations d'insertion indépendamment des attributs null de la colonne source. Dans le cas d'une mise à jour, cette colonne reflète les données qui ont été mises à jour. Dans le cas d'une suppression, cette colonne reflète les données qui ont été supprimées. Dans le cas d'une insertion, cette colonne est null.

## 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	<b>Type de données</b> : CHAR(128) ; <b>NULL admis</b> : oui  L'ID d'autorisation primaire pour la transaction que vous voulez ignorer.
AUTHTOKEN	<b>Type de données</b> : CHAR(30); <b>Null admis</b> : Oui  <b>z/OS</b> Jeton d'autorisation (nom de travail) pour la transaction que vous voulez ignorer.
PLANNAME	<b>Type de données</b> : CHAR(8) ; <b>NULL admis</b> : oui  <b>z/OS</b> Le nom de plan pour la transaction que vous voulez ignorer.
IGNTRANTRC	<b>Type de données</b> : CHAR(1) ; <b>NULL admis</b> : 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 :  <b>N (par défaut)</b> La trace est désactivée.  <b>Y</b> 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	<b>Type de données</b> : TIMESTAMP ; <b>NULL admis</b> : non, avec valeur par défaut L'heure à laquelle la transaction a été ignorée. Valeur par défaut : horodatage actuel
AUTHID	<b>Type de données</b> : CHAR(128) ; <b>NULL admis</b> : oui L'ID d'autorisation primaire de la transaction qui a été ignoré.
AUTHTOKEN	<b>Type de données</b> : CHAR(30); <b>Null admis</b> : Oui  Jeton d'autorisation (nom de travail) pour la transaction qui a été ignorée.
PLANNAME	<b>Type de données</b> : CHAR(8) ; <b>NULL admis</b> : oui  Nom de plan pour la transaction qui a été ignorée.
TRANSID	<b>Type de données</b> : CHAR(10) POUR BIT DATA ; <b>NULL admis</b> : Non. Identificateur de transaction pour la transaction qui a été ignorée.
COMMITLSN	<b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : Non Numéro de séquence du journal de validation pour la transaction qui a été ignorée.

## Table IBMSNAP\_PARTITIONINFO

La table IBMSNAP\_PARTITIONINFO complète la table IBMSNAP\_RESTART dans un environnement à plusieurs partitions et contient des informations qui permettent au programme Capture de redémarrer à partir du premier numéro d'ordre du journal nécessaire dans chaque ensemble de partitions des fichiers journaux.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : PARTITIONID,USAGE

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données. Si vous supprimez la ligne de cette table, le programme Capture est obligé de démarrer à froid.

Dans un environnement à plusieurs partitions, la table IBMSNAP\_PARTITIONINFO et la table IBMSNAP\_RESTART remplacent la table IBMSNAP\_WARM\_START de la version 7 et des versions précédentes de la réplication SQL. Une ligne est insérée dans cette table chaque fois qu'une partition

est ajoutée. Le programme Capture démarrera la lecture du fichier journal des nouvelles partitions à partir du premier numéro LSN utilisé par DB2 après la première émission de CONNECT.

Si vous n'avez jamais démarré le programme Capture, cette table est vide et le programme Capture doit effectuer un démarrage à froid.

Le tableau 71 fournit une brève description des colonnes dans la table IBMSNAP\_PARTITIONINFO.

Tableau 71. Colonnes dans la table IBMSNAP\_PARTITIONINFO

Nom de colonne	Description
PARTITIONID	<b>Type de données</b> : INT ; <b>Null admis</b> : non  ID de partition pour chaque partition valide.
USAGE	<b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non  Utilisation du numéro d'ordre du journal (LSN). Un R dans cette colonne indique que le numéro LSN a été redémarré.
SEQUENCE	<b>Type de données</b> : CHAR(10) pour données binaires ; <b>Null admis</b> : non  Numéro LSN de redémarrage pour la partition qui possède l'ID de partition.
STATUS	<b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui  Etat de la partition. Un A dans cette colonne indique que la partition est active. Cette colonne est réservée pour un usage ultérieur.
LAST_UPDATE	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : oui  Horodatage du numéro LSN de redémarrage pour la partition dont l'ID a été mis à jour en dernier.

## Table IBMSNAP\_PRUNCNTL

La table de contrôle d'élagage contient des informations détaillées relatives à tous les membres d'ensembles d'abonnements définis pour ce schéma de Capture. Cette table est utilisée conjointement avec la table IBMSNAP\_PRUNE\_SET lors de l'élagage. Elle est également utilisée lors du processus d'établissement de liaison d'initialisation entre les programmes Apply et Capture.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : SOURCE\_OWNER, SOURCE\_TABLE, SOURCE\_VIEW\_QUAL, APPLY\_QUAL, SET\_NAME, TARGET\_SERVER, TARGET\_TABLE, TARGET\_OWNER

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Pour les sources DB2, vous pouvez appeler l'élagage en émettant la commande **prune** ou en mode automatique. Pour des sources relationnelles non-DB2, l'élagage est effectué par le déclencheur d'élagage qui a été créé lorsque vous avez enregistré la source.

Le tableau 72 fournit une brève description des colonnes dans la table IBMSNAP\_PRUNCNTL.

Tableau 72. Colonnes dans la table IBMSNAP\_PRUNCNTL

Nom de colonne	Description
TARGET_SERVER	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Nom de serveur où réside la table ou la vue cible pour ce membre.
TARGET_OWNER	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non  Qualificatif de haut niveau pour la table ou la vue cible pour ce membre.
TARGET_TABLE	<b>Type de données</b> : VARCHAR(128), VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou précédente ; <b>Null admis</b> : non  Nom de la table ou de la vue cible pour ce membre.
SYNCHTIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : oui  Le programme Capture définit cet horodatage lors du processus d'établissement de liaison d'initialisation avec le programme Apply. La valeur vient de l'horodatage de l'enregistrement de journal de validation associé à la transaction de l'insertion de signal CAPSTART. Il ne sera pas remis à jour, sauf si un autre processus d'initialisation a lieu.
SYNCHPOINT	<b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui  Le programme Capture définit cette valeur lors du processus d'établissement de liaison d'initialisation avec le programme Apply. La valeur vient du numéro LSN de l'enregistrement de journal de validation associé à la transaction de l'insertion de signal CAPSTART. Il ne sera pas remis à jour, sauf si un autre processus d'initialisation a lieu.
SOURCE_OWNER	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non  Qualificatif de haut niveau de la table ou de la vue source pour ce membre.
SOURCE_TABLE	<b>Type de données</b> : VARCHAR(128), VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou précédente ; <b>Null admis</b> : non  Nom de la table ou de la vue source pour ce membre.
SOURCE_VIEW_QUAL	<b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non  Cette colonne sert à prendre en charge plusieurs enregistrements pour différentes vues sources avec des valeurs de colonnes SOURCE_OWNER et SOURCE_TABLE identiques. Cette valeur est égale à 0 pour les tables physiques définies en tant que sources et est supérieure à 0 pour les vues définies en tant que sources.
APPLY_QUAL	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Qualificatif Apply identifiant quel programme Apply traite ce membre.
SET_NAME	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Nom de l'ensemble d'abonnements auquel ce membre d'ensemble d'abonnements appartient.
CNTL_SERVER	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Nom du serveur sur lequel résident les tables de contrôle Apply pour ce programme Apply, identifié par APPLY_QUAL.

Tableau 72. Colonnes dans la table IBMSNAP\_PRUNCNTL (suite)

Nom de colonne	Description
TARGET_STRUCTURE	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non</p> <p>Valeur identifiant le type de table ou de vue cible :</p> <p>1 Table source</p> <p>3 Table de modification cohérente des données</p> <p>4 Table des points de cohérence</p> <p>5 Table d'agrégation de base</p> <p>6 Table d'agrégation des modifications</p> <p>7 Table réplique</p> <p>8 Table de copie utilisateur</p> <p>9 Table de modification cohérente des données sans jointure des tables IBMSNAP_UOW et CD</p>
CNTL_ALIAS	<p><b>Type de données</b> : CHAR(8) ; <b>Null admis</b> : oui</p> <p>Alias DB2 correspondant au serveur de contrôle Apply nommé dans la colonne CNTL_SERVER.</p>
PHYS_CHANGE_OWNER	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui</p> <p>Valeur dans la colonne PHYS_CHANGE_OWNER de la table IBMSNAP_REGISTER associée à la source de ce membre d'un ensemble d'abonnements particulier.</p>
PHYS_CHANGE_TABLE	<p><b>Type de données</b> : VARCHAR(128); VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou antérieure ; <b>Null admis</b> : oui</p> <p>Valeur dans la colonne PHYS_CHANGE_TABLE de la table IBMSNAP_REGISTER associée à la source de ce membre d'un ensemble d'abonnements particulier.</p>
MAP_ID	<p><b>Type de données</b> : VARCHAR(10) ; <b>Null admis</b> : non</p> <p>Facteur d'unicité fournissant un index plus court et plus facile à utiliser dans cette table. Il sert également à associer les insertions CAPSTART dans la table de signaux à la ligne correspondante dans la table de contrôle d'élagage.</p>

## Table IBMSNAP\_PRUNE\_LOCK

La table IBMSNAP\_PRUNE\_LOCK permet de sérialiser l'accès des tables de modification des données lors d'un démarrage à froid ou d'un élagage en fonction de la durée de conservation. Cette table permet de s'assurer que le programme Apply n'accède pas à la table de modification des données lors de ces phases critiques. Cette table ne comporte aucune ligne.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : aucun

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

## Table IBMSNAP\_PRUNE\_SET

La table IBMSNAP\_PRUNE\_SET suit le déroulement des programmes Capture et Apply pour chaque ensemble d'abonnements afin d'aider à coordonner l'élagage des tables de modification des données et des tables UOW. Contrairement à la table IBMSNAP\_PRUNCNTL, qui comporte une ligne par mappage source-vers-cible, la table IBMSNAP\_PRUNE\_SET possède une ligne par ensemble d'abonnements.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : TARGET\_SERVER, APPLY\_QUAL, SET\_NAME

**Important** : Faites attention lorsque vous mettez à jour cette table avec 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 IBMSNAP\_PRUNE\_SET.

Tableau 73. Colonnes dans la table IBMSNAP\_PRUNE\_SET

Nom de colonne	Description
TARGET_SERVER	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non Nom de serveur où réside les tables ou les vues cibles pour cet ensemble.
APPLY_QUAL	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non Qualificatif Apply identifiant quel programme Apply traite cet ensemble.
SET_NAME	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non Nom de l'ensemble d'abonnements.
SYNCHTIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : oui Le programme Apply utilise cette colonne pour enregistrer sa progression, indiquant qu'il a traité des données jusqu'à cet horodatage pour l'ensemble d'abonnements.
SYNCHPOINT	<b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : non Le programme Apply utilise cette colonne pour enregistrer sa progression, indiquant qu'il a traité des données jusqu'à cette valeur de point de synchronisation pour l'ensemble d'abonnements.

## IBMSNAP\_REG\_EXT (System i)

La table IBMSNAP\_REG\_EXT est une table spécifique à System i qui fournit des informations supplémentaires pour la table IBMSNAP\_REGISTER. Pour chaque ligne dans la table IBMSNAP\_REGISTER, il existe une ligne correspondante dans la table IBMSNAP\_REG\_EXT qui contient des colonnes supplémentaires, spécifiques à System i.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index :** VERSION, SOURCE\_OWNER, SOURCE\_TABLE, SOURCE\_VIEW\_QUAL

**Important :** prenez garde au moment de mettre à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Cette table est conservée par un programme de déclenchement (programme QZSNJLV8 dans la bibliothèque QDP4) dans la table IBMSNAP\_REGISTER. Le déclencheur est défini au moment où la table IBMSNAP\_REGISTER est créée.

Les informations de cette table sont utilisées pour tracer l'emplacement et la manière dont vous avez défini vos sources de réplication sur un serveur System i.

Le tableau 74 fournit une brève description des colonnes de la table IBMSNAP\_REG\_EXT.

Tableau 74. Colonnes de la table IBMSNAP\_REG\_EXT

Nom de colonne	Description
VERSION	<b>Type de données :</b> INT ; <b>Null admis :</b> non  La version de DB2 DataPropagator pour System i que vous avez utilisée pour enregistrer la source.
SOURCE_OWNER	<b>Type de données :</b> VARCHAR(30) ; <b>Null admis :</b> non  Qualificatif de haut niveau de la table ou de la vue source que vous avez enregistrée.
SOURCE_TABLE	<b>Type de données :</b> VARCHAR(128); <b>Null admis :</b> Non  Nom de la table ou de la vue source que vous avez enregistrée.
SOURCE_NAME	<b>Type de données :</b> CHAR(10) ; <b>Null admis :</b> oui  Nom de système à dix caractères de la table source ou de la vue que vous avez utilisée pour émettre les commandes.
SOURCE_MBR	<b>Type de données :</b> CHAR(10) ; <b>Null admis :</b> oui  Nom du membre de la table source qui est utilisé pour émettre les commandes Receive Journal Entry ( <b>RCVJRNE</b> ) et la prise en charge <b>ALIAS</b> .
SOURCE_TABLE_RDB	<b>Type de données :</b> CHAR(18) ; <b>Valeur NULL admise :</b> oui  Lors de l'utilisation de journaux distants, cette colonne contient le nom de la base de données du système où réside réellement la table source. Pour les journaux locaux, cette colonne est null.
JRN_LIB	<b>Type de données :</b> CHAR(10) ; <b>Null admis :</b> oui  Nom de bibliothèque utilisé par la table source.
JRN_NAME	<b>Type de données :</b> CHAR(10) ; <b>Null admis :</b> oui  Nom de journal utilisé par une table source. Dans cette colonne, une astérisque suivie de neufs blancs signifie qu'actuellement la table source n'est pas dans un journal ; dans ce cas, le programme Capture ne peut pas capturer de données pour cette table source.
FR_START_TIME	<b>Type de données :</b> TIMESTAMP ; <b>Null admis :</b> oui  Heure à laquelle le programme Apply a commencé à effectuer une régénération intégrale.



Tableau 74. Colonnes de la table IBMSNAP\_REG\_EXT (suite)

Nom de colonne	Description
SOURCE_VIEW_QUAL	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non</p> <p>Prend en charge la vue des abonnement en mettant en correspondance la colonne similaire dans la table de registres. Cette valeur est égale à 0 pour les tables physiques définies en tant que source et est supérieure à 0 pour les vues définies en tant que sources. Vous devez avoir cette colonne pour prendre en charge plusieurs abonnements pour différentes vues sources contenant des valeurs de colonnes SOURCE_OWNER et SOURCE_TABLE identiques.</p>
CMT_BEHAVIOR_CASE	<p><b>Type de données</b> : SMALLINT; <b>Null admis</b> : Non, avec valeur par défaut ; <b>Valeur par défaut</b> : 0</p> <p>Nombre entier représentant la manière dont les programmes d'application qui mettent à jour la table source utilisent le contrôle de validation. Le programme Capture utilise cette valeur pour gérer son utilisation de mémoire pour les lignes de données de modification qu'il a construites mais qu'il n'est pas encore prêt à écrire dans les tables de données de modification.</p> <p>-1 Le modèle de contrôle de validation n'est pas encore établi pour les applications. Il s'agit de la valeur initiale de la colonne.</p> <p>0 Aucune des applications qui mettent à jour la source n'utilise le contrôle de validation.</p> <p>1 Toutes les applications qui mettent à jour la source utilisent le contrôle de validation. C'est pourquoi deux applications différentes ne mettent jamais à jour à la même table source sous le contrôle de validation en même temps.</p> <p>2 Pour les applications simultanées qui mettent à jour la source, certaines utilisent le contrôle de validation, d'autres non. Il se peut que deux applications mettent à jour la table source en utilisant simultanément le contrôle de validation.</p>
MAX_ROWS_BTWN_CMTS	<p><b>Type de données</b> : SMALLINT; <b>Null admis</b> : Non, avec valeur par défaut ; <b>Valeur par défaut</b> : 0</p> <p>Nombre maximal de lignes que le programme Capture peut traiter avant de valider les données dans la table de données de modification.</p>

## Table IBMSNAP\_REGISTER

La table IBMSNAP\_REGISTER contient des informations sur les sources de réplication, telles que les noms des tables sources de réplication, leurs attributs et les noms des tables de modification des données et des tables CCD associées. Une ligne est automatiquement insérée dans cette table chaque fois que vous définissez une nouvelle table ou vue source de réplication pour le programme Capture à traiter.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : SOURCE\_OWNER, SOURCE\_TABLE, SOURCE\_VIEW\_QUAL

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Pour savoir comment vous avez défini vos sources de réplication, vous devez consulter la table de registres.

Le tableau 75 fournit une brève description des colonnes dans la table IBMSNAP\_REGISTER.

Tableau 75. Colonnes dans la table IBMSNAP\_REGISTER

Nom de colonne	Description																
SOURCE_OWNER	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non  Qualificatif de haut niveau de la table ou de la vue source que vous avez enregistrée.																
SOURCE_TABLE	<b>Type de données</b> : VARCHAR(128), VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou précédente; <b>Null admis</b> : non  Nom de la table ou de la vue source que vous avez enregistrée.																
SOURCE_VIEW_QUAL	<b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non  Cette colonne sert à prendre en charge plusieurs enregistrements pour différentes vues sources avec des valeurs de colonnes SOURCE_OWNER et SOURCE_TABLE identiques. Cette valeur est égale à 0 pour les tables physiques définies en tant que sources et est supérieure à 0 pour les vues définies en tant que sources.																
GLOBAL_RECORD	<b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non																
SOURCE_STRUCTURE	<b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non  Valeur identifiant la structure de table ou de vue source :  <table> <tr> <td>1</td> <td>Table utilisateur</td> </tr> <tr> <td>3</td> <td>Table de modification cohérente des données</td> </tr> <tr> <td>4</td> <td>Table des points de cohérence</td> </tr> <tr> <td>5</td> <td>Table d'agrégation de base</td> </tr> <tr> <td>6</td> <td>Table d'agrégation des modifications</td> </tr> <tr> <td>7</td> <td>Table réplique</td> </tr> <tr> <td>8</td> <td>Table de copie utilisateur</td> </tr> <tr> <td>9</td> <td>Table de modification cohérente des données sans jointure des tables IBMSNAP_UOW et CD</td> </tr> </table>	1	Table utilisateur	3	Table de modification cohérente des données	4	Table des points de cohérence	5	Table d'agrégation de base	6	Table d'agrégation des modifications	7	Table réplique	8	Table de copie utilisateur	9	Table de modification cohérente des données sans jointure des tables IBMSNAP_UOW et CD
1	Table utilisateur																
3	Table de modification cohérente des données																
4	Table des points de cohérence																
5	Table d'agrégation de base																
6	Table d'agrégation des modifications																
7	Table réplique																
8	Table de copie utilisateur																
9	Table de modification cohérente des données sans jointure des tables IBMSNAP_UOW et CD																
SOURCE_CONDENSED	<b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non  Indicateur signalant si la table source est une table condensée, ce qui signifie que toutes les lignes ayant la même clé sont condensées en une seule ligne :  <table> <tr> <td>Y</td> <td>La source est condensée.</td> </tr> <tr> <td>N</td> <td>La source n'est pas condensée.</td> </tr> <tr> <td>A</td> <td>La source est une table d'agrégation de base ou une table d'agrégation des modifications.</td> </tr> </table>	Y	La source est condensée.	N	La source n'est pas condensée.	A	La source est une table d'agrégation de base ou une table d'agrégation des modifications.										
Y	La source est condensée.																
N	La source n'est pas condensée.																
A	La source est une table d'agrégation de base ou une table d'agrégation des modifications.																

Tableau 75. Colonnes dans la table IBMSNAP\_REGISTER (suite)

Nom de colonne	Description
SOURCE_COMPLETE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Indicateur signalant comment la table source stocke les lignes de valeurs de clé primaire :</p> <p><b>Y</b> La table source contient une ligne par valeur de clé primaire présentant un intérêt.</p> <p><b>N</b> La table source contient un sous-ensemble de lignes de valeurs de clé primaire.</p>
CD_OWNER	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui</p> <p>Qualificatif de haut niveau de la table de modification des données source.</p> <p><b>Pour les tables sources</b> Pour toutes les tables sources enregistrées qui ne sont pas des tables CCD externes, cette colonne contient le qualificatif de haut niveau de la table de modification des données associée à cette table de modification des données.</p> <p><b>Pour les vues sources</b> Cette colonne contient le qualificatif de haut niveau de la vue de modification des données.</p> <p><b>Pour les tables CCD sources</b> La valeur de cette colonne est null.</p>
CD_TABLE	<p><b>Type de données</b> : VARCHAR(128), VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou précédente ; <b>Null admis</b> : oui</p> <p>Nom de la table de modification des données source.</p> <p><b>Pour les tables sources</b> Pour toutes les tables sources enregistrées qui ne sont pas des tables CCD externes, cette colonne contient le nom de la table de modification des données qui conserve les mises à jour capturées de la table source.</p> <p><b>Pour les vues sources</b> Cette colonne contient le nom de la vue de modification des données.</p> <p><b>Pour les tables CCD sources</b> La valeur de cette colonne est null.</p>
PHYS_CHANGE_OWNER	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui</p> <p>Qualificatif de haut niveau de la table ou de la vue que le programme Apply utilise pour la réplication en mode capture des modifications :</p> <p><b>Pour les tables sources</b> Pour toutes les tables sources enregistrées qui ne sont pas des tables CCD externes, cette colonne contient le qualificatif de haut niveau de la table de modification des données physique associée à cette table de modification des données.</p> <p><b>Pour les vues sources</b> Cette colonne contient le qualificatif de haut niveau de la table de modification des données physique associée à cette vue source.</p> <p><b>Pour les tables CCD sources</b> Cette colonne contient le qualificatif de haut niveau de la vue ou de la table CCD externe.</p>

Tableau 75. Colonnes dans la table IBMSNAP\_REGISTER (suite)

Nom de colonne	Description
PHYS_CHANGE_TABLE	<p><b>Type de données</b> : VARCHAR(128), VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou précédente ; <b>Null admis</b> : oui</p> <p>Nom de la table ou de la vue que le programme Apply utilise pour la réplication en mode capture des modifications :</p> <p><b>Pour les tables sources</b> Pour toutes les tables sources enregistrées qui ne sont pas des tables CCD externes, cette colonne contient le nom de la table de modification des données physique associée à cette table de modification des données.</p> <p><b>Pour les vues sources</b> Cette colonne contient le nom de la table de modification des données physique associée à cette vue source.</p> <p><b>Pour les données CCD sources</b> Cette colonne contient le nom de la vue ou de la table CCD externe.</p>
CD_OLD_SYNCHPOINT	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui</p> <p>Cette colonne est utilisée pour l'établissement de liaison initial entre le programme Apply et le programme Capture. Puis le programme Capture commence la capture des données de ce numéro LSN dans le journal source. Cette colonne sert également à montrer qu'un élagage en fonction de la durée de conservation s'est produit pour une table de modification des données. Si cette valeur est null, l'enregistrement est inactif.</p>
CD_NEW_SYNCHPOINT	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui</p> <p>Le programme Capture avance dans cette colonne au fur et à mesure qu'il insère de nouvelles lignes dans la table de modification des données. Le programme Apply utilise cette colonne pour savoir s'il existe de nouvelles modifications à répliquer.</p>
DISABLE_REFRESH	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si les régénérations intégrales sont autorisées :</p> <p>0 Les régénérations intégrales sont autorisées.</p> <p>1 Les régénérations intégrales sont empêchées.</p>
CCD_OWNER	<p><b>Type de données</b> : VARCHAR(30), VARCHAR(128) pour DB2 pour les sous-systèmes de mode nouvelle fonction z/OS version 8 ; <b>Null admis</b>:Oui</p> <p>Pour une source qui possède une table CCD interne associée, cette colonne contient le qualificatif de haut niveau des données CCD internes. Pour une table CCD externe, la valeur de cette colonne est null.</p>
CCD_TABLE	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui</p> <p>Pour une source qui possède une table CCD interne associée, cette colonne contient le nom des données CCD internes. Pour une table CCD externe, la valeur de cette colonne est null.</p>

Tableau 75. Colonnes dans la table IBMSNAP\_REGISTER (suite)

Nom de colonne	Description
CCD_OLD_SYNCHPOINT	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui</p> <p>Numéro LSN lorsque la table CCD a été réinitialisée. Cette colonne est liée au traitement de régénération intégrale par rapport aux tables CCD. La valeur de cette colonne doit être modifiée uniquement lors de la régénération initiale ou ultérieure de la table CCD. Cette valeur peut être beaucoup plus ancienne que les lignes restant dans la table CCD. Si cette colonne n'est pas gérée, le programme Apply utilisant la table CCD comme source de réplication ne saura pas que cette table a été réinitialisée et ne parviendra pas à réinitialiser les copies complètes de la source CCD.</p>
SYNCHPOINT	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui</p> <p>Dans la ligne globale (où GLOBAL_RECORD = Y), le point de synchronisation représente le numéro LSN du dernier enregistrement de journal traité par le programme Capture. Le programme conservant la table CCD avance la valeur du point de synchronisation dans toutes les lignes de la table IBMSNAP_REGISTER contenant des informations d'enregistrement sur une table CCD (interne ou externe) afin de signaler que de nouvelles données sont disponibles dans cette table CCD.</p>
SYNCHTIME	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : oui</p> <p>Dans la ligne globale (où GLOBAL_RECORD = Y), l'heure de synchronisation représente l'horodatage du dernier enregistrement de journal traité par le programme Capture. Si le programme Capture a atteint la fin du journal DB2, l'heure de synchronisation est avancée à l'horodatage DB2 actuel. Le programme conservant la table CCD avance la valeur de l'heure de synchronisation dans toutes les lignes de la table IBMSNAP_REGISTER contenant des informations d'enregistrement sur une table CCD (interne ou externe) afin de signaler le cours des données disponibles dans cette table CCD.</p>
CCD_CONDENSED	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si les données CCD internes associées à cette source sont condensées, ce qui signifie que toutes les lignes ayant la même clé sont condensées en une seule ligne :</p> <p><b>Y</b> Les données CCD internes sont condensées.</p> <p><b>N</b> Les données CCD internes ne sont pas condensées.</p> <p><b>NULL</b> Aucune table CCD interne n'est définie pour cette source.</p>
CCD_COMPLETE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si la table CCD interne associée à cette source est complète, ce qui signifie qu'elle contient initialement toutes les lignes de la table source :</p> <p><b>N</b> Les données CCD internes ne sont pas complètes.</p> <p><b>NULL</b> Aucune table CCD interne n'est définie pour cette source.</p>
ARCH_LEVEL	<p><b>Type de données</b> : CHAR(4) ; <b>Null admis</b> : non</p> <p>Niveau d'architecture des tables de contrôle de réplication :</p> <p><b>0801</b> Version 8 de la réplication SQL</p> <p><b>0803</b> Version 8 de la réplication SQL avec prise en charge des sources Oracle</p> <p><b>0805</b> Version 8 de la réplication SQL avec prise en charge de DB2 pour le mode nouvelle fonction z/OS</p>

Tableau 75. Colonnes dans la table IBMSNAP\_REGISTER (suite)

Nom de colonne	Description
DESCRIPTION	<p><b>Type de données</b> : CHAR(254) ; <b>Null admis</b> : non</p> <p>Description de la source de réplication.</p>
BEFORE_IMG_PREFIX	<p><b>Type de données</b> : VARCHAR(4) ; <b>Null admis</b> : oui</p> <p>Préfixe à un caractère identifiant les noms de colonnes image avant dans la table de modification des données. L'association du préfixe avant image et du nom de colonne de modification des données doit être sans ambiguïté, c'est-à-dire qu'un nom de colonne de modification des données ne peut pas avoir le même préfixe qu'un nom de colonne image après actuel ou possible. La longueur en octets de BEFORE_IMG_PREFIX est :</p> <p>1 Pour un caractère de préfixe simple octet ASCII ou EBCDIC.</p> <p>2 Pour un caractère de préfixe double octet ASCII.</p> <p>4 Pour un caractère de préfixe DBCS EBCDIC. Cette longueur autorise les caractères de code normal et les caractères de code spécial.</p>
CONFLICT_LEVEL	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant le niveau de détection de conflit pour cette source :</p> <p>0 Le programme Apply ne recherche pas les conflits. La cohérence des données doit être respectée par votre application afin d'éviter les éventuelles mises à jour conflictuelles.</p> <p>1 Détection standard avec rejet de transaction en cascade. Le programme Apply recherche des conflits par rapport aux modifications capturées jusque là. Le programme Apply inversera les transactions conflictuelles sur la réplique, ainsi que les transactions avec dépendances sur la transaction conflictuelle. Les modifications capturées après la début de la détection de conflit du programme Apply ne seront pas contrôlées lors de ce cycle Apply.</p> <p>2 Détection étendue avec rejet de transaction en cascade. Le programme Apply attend jusqu'à ce que le programme Apply capture toutes les modifications du journal (voir description de la colonne SYNCHTIME), puis effectue une détection standard de conflit comme pour la valeur 1. Lors de ce temps d'attente, le programme Apply maintient un verrou sur les tables sources afin de s'assurer qu'aucune modification n'est faite lors du processus de détection de conflit.</p>
CHG_UPD_TO_DEL_INS	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant comment le programme Capture stocke les mises à jour dans la table de modification des données.</p> <p>Y Le programme Capture stocke les mises à jour en utilisant deux lignes dans la table de modification des données : une pour la suppression et une pour l'insertion. Le programme Apply traite d'abord la suppression, puis l'insertion. Lorsque cet indicateur Y est défini, toutes les mises à jour d'une source de réplication sont stockées dans la table de modification des données en utilisant deux lignes. Cet indicateur permet de s'assurer que les mises à jour faites sur les colonnes de partition ou sur les colonnes enregistrées par un prédicat d'ensemble d'abonnements sont correctement traitées.</p> <p>N Toutes les mises à jour sur la table source sont stockées dans une seule ligne de la table de modification des données.</p>

Tableau 75. Colonnes dans la table IBMSNAP\_REGISTER (suite)

Nom de colonne	Description
CHGONLY	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si le programme Capture capture toutes les modifications effectuées sur la source ou uniquement les modifications effectuées dans les colonnes enregistrées. Typiquement, vous devez donner à cette option la valeur Y afin de minimiser le nombre de lignes que le programme Capture insère dans la table de modification des données, mais vous pouvez donner à cette option la valeur N afin de suivre exactement quelles lignes de la table source ont été mises à jour. Par exemple, vous pouvez capturer simplement les valeurs de la colonne de clé primaire pour contrôler quelles lignes ont été modifiées dans la table source.</p> <p><b>Y</b> Le programme Capture capture uniquement les modifications qui ont été effectuées dans les colonnes enregistrées dans le table source.</p> <p><b>N</b> Le programme Capture capture les modifications de toutes les colonnes de le table source.</p>
RECAPTURE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Cette colonne est destinée à la réplication bidirectionnelles et contient un indicateur signalant si les modifications issues d'une table ou d'une vue sont à nouveau capturées et réacheminées vers d'autres tables ou vues.</p> <p>Pour les tables sur le site maître :</p> <p><b>N</b> Les mises à jour du maître qui ont été appliquées depuis une réplique ne sont pas recapturées et ne seront pas répliquées sur d'autres répliques.</p> <p><b>Y</b> Les mises à jour du maître qui ont été appliquées depuis une réplique seront répliquées sur d'autres répliques.</p> <p>Pour les tables sur une site réplique :</p> <p><b>Y</b> Les mises à jour de la réplique qui ont été appliquées depuis le maître sont recapturées et sont disponibles pour être répliquées sur une autre table utilisant la réplique comme source.</p> <p><b>N</b> Les mises à jour de la réplique qui ont été appliquées depuis le maître ne sont pas recapturées.</p>
OPTION_FLAGS	<p><b>Type de données</b> : CHAR(4) ; <b>Null admis</b> : non</p> <p>Réservé aux options ultérieures de la réplication SQL. Actuellement, cette colonne contient la valeur par défaut de NNNN.</p>
STOP_ON_ERROR	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Oui, avec la valeur par défaut ; <b>Par défaut</b> : Y.</p> <p>Indicateur signalant si le programme Capture se terminera ou arrêtera simplement le traitement de l'enregistrement s'il rencontre des erreurs en essayant de démarrer, initialiser, réinitialiser ou insérer une ligne dans la table de modification des données :</p> <p><b>Y</b> Le programme Capture se termine si une erreur survient lorsqu'il essaie de démarrer, initialiser, réinitialiser ou insérer une ligne dans la table de modification des données.</p> <p><b>N</b> Le programme Capture arrête l'enregistrement mais ne se termine pas si une erreur survient lorsqu'il essaie de démarrer, réinitialiser ou insérer une ligne dans la table de modification des données ; il continue le traitement des autres enregistrements.</p>

Tableau 75. Colonnes dans la table IBMSNAP\_REGISTER (suite)

Nom de colonne	Description
STATE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui, avec défaut ; <b>Défaut</b> : I.</p> <p>Indicateur signalant dans quel état est l'enregistrement :</p> <p><b>S</b> Le programme Capture a arrêté le traitement de cet enregistrement. Le programme Apply ne travaillera pas avec cet enregistrement jusqu'à ce que vous réparez l'enregistrement et le mettiez en état I (inactif).</p> <p><b>A</b> L'enregistrement est actif.</p> <p><b>I</b> L'enregistrement est inactif.</p>
STATE_INFO	<p><b>Type de données</b> : CHAR(8) ; <b>Null admis</b> : oui ;</p> <p>Si le programme Capture arrête le traitement de l'enregistrement, cette colonne contient le message d'erreur qui a été émis par rapport à l'incident.</p>

## Table IBMSNAP\_REG\_SYNC (relationnelle non DB2)

La table IBMSNAP\_REG\_SYNC utilise un déclencheur UPDATE pour initier une mise à jour de la valeur SYNCHPOINT pour toutes les lignes de la table IBMSNAP\_REGISTER lorsque le programme Apply prépare l'extraction de données à partir d'une source de données relationnelles non DB2.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : TRIGGER\_ME

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 76 fournit une brève description des colonnes dans la table IBMSNAP\_REG\_SYNC.

Tableau 76. Colonnes de la table IBMSNAP\_REG\_SYNC

Nom de colonne	Description
TRIGGER_ME	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Un indicateur de Y signalant si un déclencheur a été initié pour mettre à jour la valeur SYNCHPOINT pour toutes les lignes de la table de registres.</p>
TIMESTAMP	<p>Pour les sources Microsoft SQL Server et Sybase, cette colonne contient le nombre unique généré par le système lorsqu'une mise à jour est effectuée sur une colonne d'horodatage de cette table. Cette valeur est utilisée pour dériver la valeur SYNCHPOINT enregistrée dans la table IBMSNAP_REGISTER.</p>

## Table IBMSNAP\_RESTART

La table IBMSNAP\_RESTART contient des informations qui permettent au programme Capture de redémarrer à partir du premier enregistrement de journal. Cette table remplace la table IBMSNAP\_WARM\_START de la version 7 et des versions précédentes de la réplication SQL. Elle contient une ligne qui est mise à jour à chaque point de validation ; c'est pourquoi le programme Capture peut



toujours redémarrer à l'emplacement exact sans recapturer les informations qui ont déjà été traitées et insérées dans les tables de modification des données et les tables UOW.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : aucun

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données. Si vous supprimez la ligne de cette table, le programme Capture est obligé de démarrer à froid.

Si vous n'avez jamais démarré le programme Capture, cette table est vide et le programme Capture doit effectuer un démarrage à froid.

Les deux sections suivantes montrent les agencements spécifiques aux systèmes d'exploitation de la table IBMSNAP\_RESTART.

z/OS

Linux UNIX Windows

## z/OS, Linux, UNIX, Windows

Tableau 77. Colonnes dans la table IBMSNAP\_RESTART pour z/OS, Linux, UNIX, et Windows

Nom de colonne	Description
MAX_COMMITSEQ	<b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : non  Valeur interne qui représente le point dans le journal de reprise dans lequel le programme Capture a réalisé des captures et des validations vers les tables CD et UOW.
MAX_COMMIT_TIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non  Horodatage associé à la valeur dans la colonne MAX_COMMITSEQ.
MIN_INFLIGHTSEQ	<b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : non  Valeur interne représentant le point auquel le programme Capture démarre durant un redémarrage à chaud. Cette valeur représente le premier numéro d'ordre du journal que le programme Capture a trouvé pour lequel aucun enregistrement de validation ou d'abandon n'a encore été trouvé.
CURR_COMMIT_TIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non  Horodatage actuel local auquel cette table a été mise à jour par le programme Capture.
CAPTURE_FIRST_SEQ	<b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : non  Valeur interne représentant le point à partir duquel le programme Capture a démarré au cours de son dernier démarrage à froid. Cette valeur est utilisée pour détecter si une restauration de base de données s'est produite pouvant nécessiter que le programme Capture effectue un démarrage à froid car le gestionnaire de journal de base de données peut réutiliser les numéros LSN lors de certaines opérations de restauration.

System i

## System i

Pour System i, la table IBMSNAP\_RESTART est également utilisée pour déterminer l'heure de début de la commande **RCVJRNE** (Receive Journal Entry). Une ligne est insérée dans la table de redémarrage pour chaque journal utilisé par une source de réplication ou par un groupe de sources de réplication.

**Index :** JRN\_LIB, JRN\_NAME

Tableau 78. Colonnes dans la table IBMSNAP\_RESTART pour System i

Nom de colonne	Description
MAX_COMMITSEQ	<b>Type de données :</b> CHAR(10) pour données binaires ; <b>Null admis :</b> non Numéro d'enregistrement du journal de la validation la plus courante depuis la table UOW.
MAX_COMMIT_TIME	<b>Type de données :</b> TIMESTAMP ; <b>Null admis :</b> non Horodatage associé au numéro d'enregistrement du journal dans la colonne MAX_COMMITSEQ ou horodatage actuel si le programme Capture est rattrapé avec les journaux et n'a pas de travail à effectuer.
MIN_INFLIGHTSEQ	<b>Type de données :</b> CHAR(10) pour données binaires ; <b>Null admis :</b> non Numéro LSN logique à partir duquel le programme Capture démarre lors d'un redémarrage à chaud.
CURR_COMMIT_TIME	<b>Type de données :</b> TIMESTAMP ; <b>Null admis :</b> non Horodatage actuel au moment où cette table est mise à jour.
CAPTURE_FIRST_SEQ	<b>Type de données :</b> CHAR(10) pour données binaires ; <b>Null admis :</b> non Numéro d'enregistrement du journal à partir duquel le programme Capture démarre après un démarrage à froid.
UID	<b>Type de données :</b> INTEGER ; <b>Null admis :</b> non Numéro unique utilisé comme préfixe pour le contenu de la colonne IBMSNAP_UOWID située dans la table UOW.
SEQNBR	<b>Type de données :</b> BIGINT ; <b>Null admis :</b> non Numéro de séquence de la dernière entrée de journal traitée par le programme Capture.
JRN_LIB	<b>Type de données :</b> CHAR(10) ; <b>Null admis :</b> non Nom de bibliothèque du journal que le programme Capture traite.
JRN_NAME	<b>Type de données :</b> CHAR(10) ; <b>Null admis :</b> non Nom du journal que le programme Capture traite.
STATUS	<b>Type de données :</b> CHAR(1) ; <b>Null admis :</b> oui Indicateur signalant si le programme Capture traite un travail de consignation particulier : Y Le programme Capture traite le travail de consignation. N Le programme Capture ne traite pas le travail de consignation.

## Table IBMSNAP\_SEQTABLE (Informix)

La table IBMSNAP\_SEQTABLE contient une séquence de nombres uniques que la réplication SQL utilise comme équivalent des numéros d'ordre du journal pour les tables Informix. Ces identificateurs uniques sont utilisés dans la table IBMSNAP\_REGISTER à la place des valeurs de point de synchronisation pour que le programme Capture, le programme Apply et le moniteur d'alertes de réplication puissent communiquer le point qu'ils ont ignoré lors du dernier cycle.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index à entrées uniques** : SEQ

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 79 fournit une brève description de la colonne dans la table IBMSNAP\_SEQTABLE.

Tableau 79. Colonne dans la table IBMSNAP\_SEQTABLE

Nom de colonne	Description
SEQ	Type de données : INTEGER ; Null admis : non  Numéro unique utilisé comme identificateur de journal (point de synchronisation) pour les tables Informix.

## Table IBMSNAP\_SIGNAL

La table de signaux stocke les signaux qui demandent au programme Capture d'effectuer certaines actions. Les signaux sont envoyés par vous ou par le programme Apply.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index créé sans clause UNIQUE** : SIGNAL\_TIME

Cette table contient des informations que vous pouvez mettre à jour à l'aide de SQL.

La table IBMSNAP\_SIGNAL est créée avec l'attribut DATA CAPTURE CHANGES, ce qui signifie que toutes les opérations d'insertion, de mise à jour et de suppression effectuées sur cette table sont visibles par le programme Capture comme enregistrements de journal lus depuis le journal de récupération DB2. Le programme Capture ignore tous les enregistrements de journal de mise à jour et de suppression pour la table IBMSNAP\_SIGNAL mais reconnaît tous les enregistrements de journal d'insertion de signaux créés et validés de manière valable comme "signaux" demandant son attention. Les actions effectuées par le programme Capture pour un enregistrement de journal depuis une insertion de signal dépendent de ce qui est défini dans la table IBMSNAP\_SIGNAL pour cette insertion. Les valeurs dans la table IBMSNAP\_SIGNAL fournissent les instructions relatives à l'action souhaitée au programme Capture.

Les enregistrements dans cette table avec une valeur SIGNAL\_STATE égale à C pour complet ou des enregistrements avec un horodatage admissible pour l'élagage en fonction de la durée de conservation sont supprimés quand le programme Capture réalise un élagage.

Le tableau 80 fournit une brève description des colonnes dans la table IBMSNAP\_SIGNAL.

Tableau 80. Colonnes dans la table IBMSNAP\_SIGNAL

Nom de colonne	Description
SIGNAL_TIME	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non, avec valeur par défaut ; <b>Défaut</b> : horodatage actuel.</p> <p>Un horodatage utilisé pour identifier de façon unique la ligne. Le programme Capture utilise cette valeur unique pour trouver la ligne correcte dans la table des signaux indiquant la fin du traitement du signal Capture. Cette colonne d'horodatage est créée comme non NULL avec défaut. Un signal de Capture peut donc généralement être inséré de façon à ce que DB2 fournisse l'horodatage actuel comme valeur SIGNAL_TIME.</p>
SIGNAL_TYPE	<p><b>Type de données</b> : VARCHAR(30) ; <b>Null admis</b> : non</p> <p>Indicateur signalant le type de signal qui a été posté :</p> <p><b>CMD</b> Signal posté par vous, le programme Apply ou une autre application, qui est une commande système ou un signal connu. Voir la colonne SIGNAL_SUBTYPE de cette table pour une liste des sous-types de signaux disponibles.</p> <p><b>USER</b> Signal posté par vous ou par un autre utilisateur. Le programme Capture met à jour la valeur de la colonne SIGNAL_LSN avec le numéro LSN issu du journal au moment où le signal a été inséré et met à jour la valeur de la colonne SIGNAL_STATE en la passant de P (en attente) à R (reçu).</p>

Tableau 80. Colonnes dans la table IBMSNAP\_SIGNAL (suite)

Nom de colonne	Description
SIGNAL_SUBTYPE	<p><b>Type de données</b> : VARCHAR(30) ; <b>Null admis</b> : oui</p> <p>Action que le programme Capture effectue lorsqu'un signal provenant d'une commande système (SIGNAL_TYPE = CMD) survient.</p> <p><b>CAPSTART</b> Le programme Capture démarre la capture des modifications sur la source enregistrée pour un membre d'un ensemble d'abonnements particulier identifié par MAP_ID (de la table IBMSNAP_PRUNCNTL) dans la colonne SIGNAL_INPUT_IN. Par exemple, le programme Apply émet ce signal avant d'effectuer une régénération intégrale de toutes les tables cibles de l'ensemble afin de faire savoir au programme Capture que l'ensemble est prêt à commencer la réplication en mode capture des modifications. Le programme Apply poste ce signal.</p> <p><b>STOP</b> Le programme Capture arrête la capture des modifications et se termine. Cette commande ne peut être émise que par vous, pas par le programme Apply.</p> <p><b>CAPSTOP</b> Le programme Capture arrête de capturer les modifications pour une source enregistrée particulière identifiée par <i>propriétaire_source.table_source</i> dans la colonne SIGNAL_INPUT_IN. Cette commande ne peut être émise que par vous, pas par le programme Apply.</p> <p><b>UPDANY</b> Le programme Apply (identifié par le qualificatif Apply dans la colonne SIGNAL_INPUT_IN) fait savoir au programme Capture qu'il travaille avec deux programmes Capture dans une configuration de réplication bidirectionnelle. Le programme Apply poste ce signal. Lorsque le type de signal est USER, le sous-type de signal n'est pas utilisé ou reconnu par le programme Capture et par conséquent n'est pas une zone obligatoire. Vous pouvez définir la valeur que vous souhaitez.</p>
SIGNAL_INPUT_IN	<p><b>Type de données</b> : VARCHAR(500) ; <b>Null admis</b> : oui</p> <p>Si SIGNAL_TYPE=USER, la colonne contient une entrée définie par l'utilisateur. Si SIGNAL_TYPE = CMD, la signification de cette valeur dépend de la valeur SIGNAL_SUBTYPE pour ce signal :</p> <p><b>CMD + CAPSTART</b> Identificateur de mappage. Comme les déclencheurs de capture, et non le programme Capture, traitent les sources relationnelles non-DB2, il existe un déclencheur appelé SIGNAL_TRIGGER qui se déclenche après la mise à jour de la table IBMSNAP_SIGNAL et qui met à jour la table IBMSNAP_PRUNCNTL avec la valeur suivante de la séquence.</p> <p><b>CMD + UPDANY</b> Qualificatif Apply identifiant le programme Apply dans la configuration de réplication bidirectionnelle.</p> <p><b>CMD + CAPSTOP</b> Nom du propriétaire de la source et table source pour laquelle le programme Capture doit arrêter la capture des modifications pour (<i>propriétaire_source.table_source</i>).</p>

Tableau 80. Colonnes dans la table IBMSNAP\_SIGNAL (suite)

Nom de colonne	Description
SIGNAL_STATE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Indicateur signalant l'état du signal :</p> <p><b>P</b> Le signal est en attente ; le programme Capture ne l'a pas encore reçu. Lorsque vous postez un signal, définissez la valeur de SIGNAL_STATE à P.</p> <p><b>R</b> Le programme Capture a reçu le signal. Le programme Capture définit la valeur SIGNAL_STATE à R (au lieu de la modifier en C pour complet) lorsqu'il reçoit un signal où SIGNAL_TYPE = USER ou un signal où SIGNAL_TYPE = CMD et SIGNAL_SUBTYPE = STOP.</p> <p><b>C</b> Le programme Capture a terminé le traitement du signal. Le programme Capture définit cette valeur à C lorsque SIGNAL_TYPE = CMD pour toutes les valeurs SIGNAL_SUBTYPE à l'exception de STOP.</p>
SIGNAL_LSN	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui</p> <p>Numéro LSN de l'enregistrement de validation. Cette valeur est définie uniquement par le programme Capture.</p>

**System i** Sur les systèmes d'exploitation System i, une table de signaux est associée à chaque journal utilisé pour les tables sources. Ces tables sont appelées tables de signaux de journal et ont la même structure que la table globale IBMSNAP\_SIGNAL. Le nom de la table de signaux de journal est *schéma*.IBMSNAP\_SIGNAL\_XXXX\_yyyy, où XXXX est la bibliothèque de journal et yyyy est le nom de journal. Cette table est créée automatiquement et est consignée dans le journal source sur le serveur source.

## Table IBMQREP\_TABVERSION

La table IBMQREP\_TABVERSION est utilisée par les programmes Q Capture et Capture pour conserver une trace des différentes versions d'une 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 81, à la page 451 fournit une brève description des colonnes de la table IBMQREP\_TABVERSION.

Tableau 81. Colonnes de la table IBMQREP\_TABVERSION

Nom de colonne	Description
LSN	<b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : Non  Point du journal de récupération DB2 où le programme Q Capture ou Capture a détecté une nouvelle version de la table source.
TABLEID1	<b>Type de données</b> : SMALLINT ; <b>NULL admis</b> : non  Identificateur de base de données (DBID) dans SYSIBM.SYSTABLES.
TABLEID2	<b>Type de données</b> : SMALLINT ; <b>NULL admis</b> : non  Identificateur d'objet (OBID) dans SYSIBM.SYSTABLES.
VERSION	<b>Type de données</b> : INTEGER ; <b>NULL admis</b> : Non  Nombre généré par le programme Q Capture ou Capture pour conserver une trace des différentes versions d'une table source.
SOURCE_OWNER	<b>Type de données</b> : VARCHAR(128) ; <b>NULL admis</b> : non  Schéma ou qualificatif de haut niveau de la table source.
SOURCE_NAME	<b>Type de données</b> : VARCHAR(128) ; <b>NULL admis</b> : non  Nom de la table source.

## Table IBMSNAP\_UOW

La table IBMSNAP\_UOW fournit des informations supplémentaires sur les transactions qui ont été validées sur une table source. Pour tous les types de table cible différents d'une copie utilisateur, le programme Apply joint IBMSNAP\_UOW et les tables de modification des données (CD) en fonction des valeurs IBMSNAP\_COMMITSEQ correspondantes lorsqu'il applique des modifications aux tables cibles. Si vous démarrez à froid le programme Capture, toutes ces entrées dans cette table sont supprimées.

**Serveur** : serveur de contrôle Capture

**Schéma par défaut** : ASN

**Index** : IBMSNAP\_COMMITSEQ, IBMSNAP\_LOGMARKER

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

### System i

- Capture pour System i pouvant démarrer la capture de données pour un sous-ensemble de sources de réplication, il ne supprime pas toutes les lignes de la table IBMSNAP\_UOW si vous faites un démarrage à froid partiel.
- Certains programmes utilisateur n'utilisent pas le contrôle de validation. Dans ce cas, Capture pour System i insère arbitrairement une nouvelle ligne UOW après l'écriture d'un certain nombre de lignes dans la table de modification des données. Cette frontière de validation artificielle permet de réduire la taille de la table UOW.
- La table UOW est élaguée par les durées de conservation, pas par les informations provenant de la table IBMSNAP\_PRUNE\_SET.

Le programme Capture nécessite une table IBMSNAP\_UOW par schéma de Capture. Le programme Capture insère une nouvelle ligne dans cette table pour chaque enregistrement de journal validé sur la source de réplication.

Le programme Capture élague également la table UOW en fonction des informations que le programme Apply insère dans la table IBMSNAP\_PRUNE\_SET.

Le tableau 82 fournit une brève description des colonnes dans la table IBMSNAP\_UOW.

Tableau 82. Colonnes dans la table IBMSNAP\_UOW

Nom de colonne	Description
IBMSNAP_UOWID	<p><b>Type de données :</b> VARCHAR(16) FOR BIT DATA ; <b>Null admis :</b> non</p> <p>Identificateur UOW de l'en-tête d'enregistrement de journal pour cette unité de travail. Vous pouvez définir que cette colonne fait partie d'une table cible CCD incomplète.</p>
IBMSNAP_COMMITSEQ	<p><b>Type de données :</b> VARCHAR(16) FOR BIT DATA ; <b>Null admis :</b> non</p> <p>Le numéro de séquence d'enregistrement de journal de l'instruction de validation enregistrée. Pour tous les types de table cible différents d'une copie utilisateur, le programme Apply joint les tables de modification des données et les tables UOW en fonction des valeurs dans cette colonne lorsqu'il applique des modifications aux tables cibles.</p>
IBMSNAP_LOGMARKER	<p><b>Type de données :</b> TIMESTAMP ; <b>Null admis :</b> non</p> <p>L'heure approximative (sur le serveur de contrôle Capture) à laquelle les données ont été validées.</p>
IBMSNAP_AUTHTKN	<p><b>Type de données :</b> VARCHAR(30) ; <b>Null admis :</b> non</p> <p>Clé d'accès qui est associée à la transaction. Cet ID est utile pour l'audit de la base de données. Pour DB2 for z/OS, cette colonne est l'identificateur de corrélation. Pour DB2 pour i5/OS, cette colonne est le nom du travail qui a causé une transaction. Cette colonne n'est pas copiée automatiquement dans les autres tables ; vous devez la sélectionner et la copier en tant que colonne de données utilisateur. Vous pouvez définir que cette colonne fait partie d'une table cible CCD incomplète.</p>
IBMSNAP_AUTHID	<p><b>Type de données :</b> VARCHAR(128) ; <b>Null admis :</b> non</p> <p>Identificateur d'autorisation associé à la transaction. Cette colonne est utile pour l'audit de la base de données. Pour DB2 for z/OS, cette colonne est l'ID utilisateur primaire. Pour DB2 pour i5/OS, cette colonne a le nom de l'ID de profil utilisateur sous lequel l'application qui a causé la transaction a été exécutée. Cette colonne conserve l'ID de dix caractères complété avec des blancs. Cette colonne n'est pas copiée automatiquement dans les autres tables ; vous devez la sélectionner et la copier en tant que colonne de données utilisateur. Vous pouvez définir cette colonne de façon à ce qu'elle fasse partie d'une table cible CCD incomplète.</p>



Tableau 82. Colonnes dans la table IBMSNAP\_UOW (suite)

Nom de colonne	Description
IBMSNAP_REJ_CODE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non, avec défaut ; <b>Défaut</b> : 0.</p> <p>Indicateur signalant si des lignes ont été rejetés et annulées. Cette valeur est définie uniquement lors de la réplication bidirectionnelle si la détection de conflit est définie comme standard ou étendue lorsque vous définissez votre source de réplication. Vous pouvez définir que cette colonne fait partie d'une table cible CCD incomplète.</p> <p>0      Aucun conflit connu ne s'est produit dans la transaction.</p> <p>1      Un conflit s'est produit car une ligne identique dans le maître et la réplique a été mises à jour. La valeur de WHOS_ON_FIRST dans les tables de contrôle Apply est F. La transaction sur la réplique a été rejetée et annulée.</p> <p>2      La transaction a été rejetée et annulée car elle était dépendante d'une transaction précédente qui a été rejetée. La valeur de WHOS_ON_FIRST dans les tables de contrôle Apply est F. La transaction précédente a été rejetée car une ligne identique dans le maître et la réplique a été mise à jour et la transaction sur la réplique a été rejetée et annulée.</p> <p>3      La transaction a été rejetée et annulée car elle contenait au moins une violation de contrainte d'intégrité référentielle. Etant donné que cette transaction ne respecte pas les contraintes référentielles définies sur la table source, le programme Apply marque cet ensemble d'abonnements comme étant en échec. Les mises à jour ne peuvent pas être copiées tant que vous n'avez pas corrigé les définitions d'intégrité référentielle.</p> <p>4      La transaction a été rejetée et annulée car elle était dépendante d'une transaction précédente qui a été rejetée. La transaction précédente a été rejetée car elle contenait au moins une violation de contrainte d'intégrité référentielle.</p> <p>5      Un conflit s'est produit car une ligne identique dans le maître et la réplique a été mises à jour. La valeur de WHOS_ON_FIRST dans les tables de contrôle Apply est S. La transaction sur la réplique a été rejetée et annulée.</p> <p>6      La transaction a été rejetée et annulée car elle était dépendante d'une transaction précédente qui a été rejetée. La valeur de WHOS_ON_FIRST dans les tables de contrôle Apply est S. La transaction précédente a été rejetée car une ligne identique dans le maître et la réplique a été mise à jour et la transaction sur la réplique a été rejetée et annulée.</p>
IBMSNAP_APPLY_QUAL	<p><b>Type de données</b> : CHAR(18) ; <b>Null admis</b> : non, avec défaut ; <b>Défaut</b> : nom d'utilisateur actuel.</p> <p>Qualificatif Apply identifiant quel programme Apply a appliqué les modifications. Vous pouvez définir que cette colonne fait partie d'une table cible CCD incomplète.</p>

## Tables sur le serveur de contrôle Apply

Les tables stockées sur le serveur de contrôle Apply renferment des informations sur les définitions de vos abonnements. Pour Linux, UNIX, Windows et z/OS, vous générez ces tables de contrôle en fonction de vos spécifications via le programme de ligne de commande ASNCLP ou le Centre de réplication. Pour System i, ces tables de contrôle sont créées automatiquement à l'installation de DataPropagator for System i.

Le tableau 83 décrit les tables de contrôle sur le serveur Apply.

Tableau 83. Tables de contrôle sur le serveur Apply

Nom de la table	Description
«Table ASN.IBMSNAP_APPENQ»	Garantit qu'un seul programme Apply s'exécute pour chaque qualificatif Apply.
«Table ASN.IBMSNAP_APPLEVEL», à la page 455	Stocke la version des tables de contrôle Apply.
<b>System i</b> Table ASN.IBMSNAP_APPLY_JOB (System i)	Garantit qu'il existe un qualificatif Apply unique pour chaque instance du programme Apply s'exécutant sur un serveur de contrôle Apply.
«Table ASN.IBMSNAP_APPLYMON», à la page 456	Renferme des informations sur l'état du programme Apply.
«Table ASN.IBMSNAP_APPLYTRACE», à la page 461	Renferme des messages importants du programme Apply.
«Table ASN.IBMSNAP_APPLYTRAIL», à la page 462	Renferme des informations de trace de contrôle sur le programme Apply.
«Table ASN.IBMSNAP_APPPARMS», à la page 457	Contient des paramètres modifiables pour contrôler les opérations du programme Apply.
«Table ASN.IBMSNAP_FEEDETL», à la page 468	Identifie les membres d'un ensemble d'abonnements de la réplication SQL utilisés par InfoSphere DataStage pour alimenter un entrepôt de données.
«Table ASN.IBMSNAP_SUBS_COLS», à la page 469	Mappe des colonnes dans la table ou vue cible vers les colonnes correspondantes dans la table ou vue source.
«Table ASN.IBMSNAP_SUBS_EVENT», à la page 471	Contient des événements que vous définissez pour contrôler à quel moment le programme Apply traite un ensemble d'abonnements.
«Table ASN.IBMSNAP_SUBS_MEMBR», à la page 472	Identifie une paire de tables source et cible et indique les informations de traitement la concernant.
«Table ASN.IBMSNAP_SUBS_SET», à la page 476	Renferme des informations de traitement pour chaque série de membres d'ensemble d'abonnements que le programme Apply traite comme un groupe.
«Table ASN.IBMSNAP_SUBS_STMTS», à la page 483	Contient des appels de procédures stockées ou instructions SQL que vous définissez pour un ensemble d'abonnements. L'appel a lieu avant ou après le traitement de l'ensemble par le programme Apply.

## Table ASN.IBMSNAP\_APPENQ

La table de mise en file d'attente d'Apply sert à garantir qu'un seul programme Apply s'exécute pour chaque qualificatif Apply. Le programme Apply verrouille une seule ligne de cette table jusqu'à son arrêt. Cette table n'est pas utilisée dans System i.

**Serveur** : serveur de contrôle Apply

**Index** : APPLY\_QUAL

**Important :** Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 84 offre une description courte de la colonne dans la table IBMSNAP\_APPENQ.

Tableau 84. Colonne dans la table IBMSNAP\_APPENQ

Nom de colonne	Description
APPLY_QUAL	<b>Type de données :</b> CHAR(18) ; <b>Valeur NULL admise :</b> oui  Identifie de façon unique un groupe d'ensembles d'abonnements traités par le même programme Apply. Cette valeur fait la distinction entre les majuscules et les minuscules. Vous devez l'indiquer lorsque vous définissez un ensemble d'abonnements.

## Table ASN.IBMSNAP\_APPLEVEL

La table IBMSNAP\_APPLEVEL stocke la version des tables de contrôle Apply. La valeur stockée ici est utilisée avec la valeur des colonnes ARCH\_LEVEL et COMPATIBILITY dans la table IBMSNAP\_CAPPARMS pour déterminer la longueur des numéros de séquence de journal utilisés dans les tables Capture et Apply.

**Serveur :** serveur de contrôle Apply

tableau 85 présente une brève description de la colonne dans la table IBMSNAP\_APPLEVEL.

Tableau 85. Colonne dans la table IBMSNAP\_APPLEVEL

Nom de colonne	Description
ARCH_LEVEL	<b>Type de données :</b> CHAR(4); <b>Null admis :</b> non, valeur par défaut  La version des tables de contrôle Apply. Dans la version 10.1 sous Linux, UNIX et Windows, la valeur par défaut est 1001.

## ASN.IBMSNAP\_APPLY\_JOB (System i)

La table IBMSNAP\_APPLY\_JOB, qui est spécifique à System i, est utilisée pour garantir qu'une valeur APPLY\_QUAL unique pour toutes les instances du programme Apply s'exécute sur le serveur de contrôle Apply. Chaque fois qu'une instance du programme Apply est démarrée, une ligne est ajoutée à cette table. Si vous démarrez une nouvelle instance du programme Apply avec une valeur APPLY\_QUAL qui existe déjà, la commande de démarrage échoue.

**Serveur :** serveur de contrôle Apply

**Index :** aucun

**Important :** prenez garde au moment de mettre à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 86, à la page 456 donne une brève description des colonnes de la table IBMSNAP\_APPLY\_JOB.

Tableau 86. Colonnes de la table IBMSNAP\_APPLY\_JOB

Nom de colonne	Description
APPLY_QUAL	<p><b>Type de données :</b> CHAR(18); <b>Valeur NULL admise :</b> non</p> <p>Identifiant unique pour un groupe d'ensembles d'abonnements. Cette valeur est fournie par l'utilisateur lors de la définition d'un ensemble d'abonnements. Chaque instance du programme Apply est démarrée avec la valeur APPLY_QUAL. Cette valeur est utilisée pendant une réplication update-anywhere afin d'éviter la réplication en boucle des modifications effectuées par le programme Apply.</p>
CONTROL_SERVER	<p><b>Type de données :</b> CHAR(18); <b>Valeur NULL admise :</b> non</p> <p>Nom de la base de données dans laquelle les tables de contrôle et vues Apply sont définies.</p>
JOB_NAME	<p><b>Type de données :</b> CHAR(10) ; <b>Null admis :</b> non</p> <p>Nom qualifié complet du travail qui a écrit cette entrée de trace :</p> <p><b>Position 1-10</b> APPLY_QUAL</p> <p><b>Position 11-20</b> ID de l'utilisateur qui a démarré le programme Apply</p> <p><b>Position 21-26</b> Numéro de travail</p>
USER_NAME	<p><b>Type de données :</b> CHAR(10) ; <b>Null admis :</b> non</p> <p>Nom de l'utilisateur qui a démarré une nouvelle instance du programme Apply.</p>
JOB_NUMBER	<p><b>Type de données :</b> CHAR(6); <b>Null admis :</b> Non</p> <p>Le numéro de travail du travail en cours pour un journal particulier. Si le journal n'est pas actif, cette colonne contient le numéro de travail du dernier travail qui a été traité.</p>

## Table ASN.IBMSNAP\_APPLYMON

La table IBMSNAP\_APPLYMON contient des informations sur l'état du programme apply, y compris les ensembles d'abonnements et les membres d'un ensemble d'abonnements qu'Apply traite actuellement.

**Serveur :** serveur de contrôle Apply

**Index créé sans clause UNIQUE :** MONITOR\_TIME, APPLY\_QUAL, WHOS\_ON\_FIRST

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 87 fournit une brève description des colonnes de la table IBMSNAP\_APPLYMON.

Tableau 87. Colonnes dans la table IBMSNAP\_APPLYMON

Nom de colonne	Description
MONITOR_TIME	<p><b>Type de données :</b> TIMESTAMP ; <b>Null admis :</b> non</p> <p>Horodatage du serveur de contrôle Apply lors de l'insertion de la ligne la plus récente des informations d'état Apply.</p>

Tableau 87. Colonnes dans la table IBMSNAP\_APPLYMON (suite)

Nom de colonne	Description
APPLY_QUAL	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Qualificatif Apply auquel cette ligne d'informations d'état Apply est rattachée.
WHOS_ON_FIRST	<b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui  Indicateur du type d'abonnement que le programme Apply traite actuellement.  F Mise à jour depuis n'importe quel point. La table source est la réplique et la table cible est la table maître.  S La table source est le maître et la cible est une table autre que maître. Il peut s'agir d'un abonnement bidirectionnel ou en lecture seule.
STATE	<b>Type de données</b> : SMALLINT ; <b>Null admis</b> : oui  Etat du programme Apply :  0 Extraction depuis la source  1 Application à la cible  2 Mise en veille  3 Lecture des tables de contrôle  4 Mise à jour des tables de contrôle
CURRENT_SETNAME	<b>Type de données</b> : CHAR(18) ; <b>Valeur NULL admise</b> : oui  Ensemble traité par Apply. Cette colonne contient une valeur NULL si Apply ne traite pas un ensemble d'abonnements.
CURRENT_TABOWNER	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui  Propriétaire de la table cible ou source gérée par Apply. Cette colonne contient une valeur nulle si Apply lit ou met à jour des tables de contrôle, se met en veille ou applique les enregistrements à des tables cibles dans l'ordre transactionnel.
CURRENT_TABNAME	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui  Table cible ou source gérée par Apply. Cette colonne contient une valeur nulle si Apply lit ou met à jour une table de contrôle, se met en veille ou applique les enregistrements à des tables cibles dans l'ordre transactionnel.

## Table ASN.IBMSNAP\_APPPARMS

La table IBMSNAP\_APPPARMS renferme des paramètres modifiables pour contrôler les opérations du programme Apply. Vous pouvez définir ces paramètres afin de choisir des valeurs comme le nom du serveur de contrôle Apply sur lequel figurent les définitions d'abonnements et les tables de contrôle du programme Apply. Si vous modifiez les paramètres dans cette table, le programme Apply lit uniquement ces modifications au démarrage.

**Serveur** : serveur de contrôle Apply

**Index** : APPLY\_QUAL

Cette table contient des informations que vous pouvez mettre à jour à l'aide de SQL.

Le tableau 88 offre une description courte des colonnes dans la table IBMSNAP\_APPPARMS.

Tableau 88. Colonnes dans la table IBMSNAP\_APPPARMS

Nom de colonne	Description
APPLY_QUAL	<p><b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non</p> <p>Le qualificatif Apply fait correspondre les paramètres au programme Apply auquel ils s'appliquent.</p>
APPLY_PATH	<p><b>Type de données</b> : VARCHAR(1040) ; <b>Valeur NULL admise</b> : Oui</p> <p>Emplacement des fichiers de travail utilisés par le programme Apply. Il s'agit par défaut du répertoire d'où le programme a été démarré.</p>
CAF	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur Null admise</b> : Oui <b>Valeur par défaut</b> : Y</p> <p>Indicateur signalant si le programme Apply s'exécute avec la connexion Call Attach Facility (CAF).</p> <p><b>Y (par défaut)</b> Le programme Apply remplace la connexion Recoverable Resource Manager Services (RRS) par défaut et s'exécute avec la connexion CAF.</p> <p><b>N</b> Le programme Apply s'exécute avec la connexion RRS.</p>
COPYONCE	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : N.</p> <p>Un indicateur signale si le programme Apply exécute un cycle de copie pour chaque ensemble d'abonnements admissible au moment où le programme est appelé.</p> <p><b>Y</b> Le programme Apply exécute un cycle de copie pour chaque ensemble d'abonnements éligible.</p> <p><b>N</b> Le programme Apply n'exécute pas un cycle de copie pour chaque ensemble d'abonnements éligible.</p>
DELAY	<p><b>Type de données</b> : INT ; <b>Valeur NULL admise</b> : Oui, avec la valeur par défaut ; <b>Par défaut</b> : 6.</p> <p>Le temps d'attente (en secondes) à la fin de chaque cycle du programme Apply lorsqu'une réplication en continu est réalisée. Ce paramètre est ignoré si vous avez spécifié <b>copyonce</b>.</p>
ERRWAIT	<p><b>Type de données</b> : INT ; <b>Valeur NULL admise</b> : Oui, avec la valeur par défaut ; <b>Par défaut</b> : 300.</p> <p>Nombre de secondes (entre 1 et 300) que le programme Apply attend avant une nouvelle tentative s'il a détecté un cas d'erreur. Ce paramètre est ignoré si vous avez spécifié <b>copyonce</b>.</p>
INAMSG	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : Y.</p> <p>Indicateur signalant si le programme Apply émet un message en cas d'inactivité.</p> <p><b>Y</b> Le programme Apply émet un message lorsqu'il est inactif.</p> <p><b>N</b> Le programme Apply n'émet pas de message lorsqu'il est inactif.</p>

Tableau 88. Colonnes dans la table IBMSNAP\_APPPARMS (suite)

Nom de colonne	Description
LOADXIT	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : N.</p> <p>Indicateur signalant si le programme Apply appelle la routine d'exit fournie par IBM (ASNLOAD) recourant aux utilitaires d'exportation et de chargement pour régénérer les tables cible.</p> <p>Y Le programme Apply appelle ASNLOAD.</p> <p>N Le programme Apply n'appelle pas ASNLOAD.</p>
LOGREUSE	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : N.</p> <p>Indicateur signalant si le programme Apply écrase le fichier journal Apply ou y ajoute des données.</p> <p>Y Le programme Apply réutilise le fichier journal en le supprimant, puis en le créant à nouveau à son redémarrage.</p> <p>N Le programme Apply ajoute de nouvelles informations au fichier journal Apply.</p>
LOGSTDOUT	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : N.</p> <p>Indicateur signalant à quel endroit le programme Apply envoie les messages du fichier journal :</p> <p>Y Le programme Apply envoie les messages du fichier journal à la fois à la sortie standard (STDOUT) et au fichier journal.</p> <p>N Le programme Apply envoie la plupart des messages de fichier journal dans le fichier journal uniquement. Les messages d'initialisation vont à la fois à la sortie standard (STDOUT) et au fichier journal.</p>
MONITOR_ENABLED	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Indicateur signalant si le programme Apply effectue des insertions dans la table IBMSNAP_APPLYMON pour enregistrer son état :</p> <p>Y Le programme Apply effectue des insertions dans IBMSNAP_APPLYMON à partir d'une planification basée sur la valeur de la colonne MONITOR_INTERVAL.</p> <p><b>N (par défaut)</b> Le programme Apply n'effectue pas d'insertions dans IBMSNAP_APPLYMON.</p>
MONITOR_INTERVAL	<p><b>Type de données</b> : INTEGER; <b>Valeur NULL admise</b> : non, avec valeur par défaut ; <b>Défaut</b> : 60000 millisecondes (1 minute)</p> <p>Fréquence, en millisecondes, à laquelle le programme Apply ajoute une ligne à la table IBMSNAP_APPLYMON.</p>
NOTIFY	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : N.</p> <p>Indicateur signalant si le programme Apply doit appeler la routine d'exit (ASNDONE) vous redonnant le contrôle quand il a terminé de copier un ensemble d'abonnements.</p> <p>Y Le programme Apply appelle ASNDONE.</p> <p>N Le programme Apply n'appelle pas ASNDONE.</p>

Tableau 88. Colonnes dans la table IBMSNAP\_APPPARMS (suite)

Nom de colonne	Description
OPT4ONE	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : N.</p> <p>Indicateur signalant si les performances du programme Apply sont optimisées si un seul ensemble d'abonnements est défini pour ce programme.</p> <p><b>Y</b> Les performances du programme Apply sont optimisées pour un ensemble d'abonnements.</p> <p><b>N</b> Les performances du programme Apply ne sont pas optimisées pour un ensemble d'abonnements.</p> <p>Ce paramètre est ignoré si vous avez spécifié <b>copyonce</b>.</p>
SLEEP	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : Y.</p> <p>Indicateur signalant comment le programme Apply doit procéder si aucun ensemble d'abonnements ne peut être traité :</p> <p><b>Y</b> Le programme Apply est mis en veille.</p> <p><b>N</b> Le programme Apply est arrêté.</p> <p>Ce paramètre est ignoré si vous avez spécifié <b>copyonce</b>.</p>
SQLERRCONTINUE	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : N.</p> <p>Indicateur signalant si le programme Apply poursuit le traitement après la recherche d'erreurs dans le fichier SQLSTATE.</p> <p><b>Y</b> Le programme Apply recherche dans le fichier SQLSTATE des erreurs SQL éventuelles pendant le traitement. Si une erreur est détectée, Apply arrête le traitement.</p> <p><b>N</b> Le programme Apply ne vérifie pas le fichier SQLSTATE et poursuit le traitement.</p>
SPILLFILE	<p><b>Type de données</b> : VARCHAR(10) ; <b>Valeur NULL admise</b> : Oui, avec la valeur par défaut.</p> <p>Indicateur signalant où la réponse extraite est stockée.</p> <p><b>z/OS</b> Les valeurs valides sont :</p> <p><b>mem (par défaut)</b> Un fichier mémoire. Si la quantité de mémoire est insuffisante pour l'ensemble de réponses, le programme Apply utilise un fichier disque.</p> <p><b>disk</b> Un fichier disque.</p> <p><b>Linux UNIX Windows</b> Les valeurs valides sont :</p> <p><b>disk (par défaut)</b> Un fichier disque.</p>



Tableau 88. Colonnes dans la table IBMSNAP\_APPPARMS (suite)

Nom de colonne	Description
TERM	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : Y.</p> <p>Indicateur signalant si le programme Apply s'arrête lorsqu'il ne peut pas se connecter à son serveur de contrôle.</p> <p><b>Y (par défaut)</b> Par défaut, le programme Apply s'arrête s'il ne peut pas se connecter à son serveur de contrôle.</p> <p><b>N</b> Le programme Apply ne s'arrête pas. Il consigne une erreur, attend le temps défini par le paramètre <b>errwait</b> et réessaie de se connecter.</p> <p>Ce paramètre est ignoré si vous avez spécifié <b>copyonce</b>.</p>
TRLREUSE	<p><b>Type de données</b> : CHAR(1) ; <b>Valeur NULL admise</b> : Yes, avec la valeur par défaut ; <b>Par défaut</b> : N.</p> <p>Indicateur signalant si le programme Apply appelle la routine d'exit fournie par IBM (ASNLOAD) recourant aux utilitaires d'exportation et de chargement pour régénérer les tables cible :</p> <p><b>Y</b> Le programme Apply appelle ASNLOAD.</p> <p><b>y</b> Le programme Apply n'appelle pas ASNLOAD.</p>
REFRESH_COMMIT_CNT	<p><b>Type de données</b> : INTEGER; <b>Valeur Null admise</b> : oui, avec la valeur par défaut ; <b>Par défaut</b> : null</p> <p>Indique le nombre de lignes que le programme Apply insère dans la table cible avant d'émettre une instruction COMMIT. Les valeurs peuvent être comprises entre 0 et 134217727. Avec une valeur par défaut nulle ou égale à 0, une seule validation est réalisée après l'insertion de toutes les lignes ; aucune validation intermédiaire n'a lieu. Cette option n'est pas prise en charge pour les sources CCD.</p>

## Table ASN.IBMSNAP\_APPLYTRACE

La table IBMSNAP\_APPLYTRACE contient des messages provenant du programme Apply. Le programme Apply ne supprime pas automatiquement cette table, mais vous pouvez rendre l'élagage automatique en ajoutant une instruction SQL qui s'exécute après un des ensembles d'abonnements.

**Serveur** : serveur de contrôle Apply

**Index** : APPLY\_QUAL, TRACE\_TIME

Le tableau 89 fournit une brève description de la colonne dans la table IBMSNAP\_APPLYTRACE.

Tableau 89. Colonnes dans la table IBMSNAP\_APPLYTRACE

Nom de colonne	Description
APPLY_QUAL	<p><b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non</p> <p>Identifie uniquement quel programme Apply a inséré le message.</p>
TRACE_TIME	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non</p> <p>L'heure sur le serveur de contrôle Apply à laquelle la ligne a été insérée dans cette table.</p>

Tableau 89. Colonnes dans la table IBMSNAP\_APPLYTRACE (suite)

Nom de colonne	Description
OPERATION	<b>Type de données</b> : CHAR(8) ; <b>Null admis</b> : non  Le type d'opération de programme Apply, par exemple initialisation, appliquer ou erreur.
DESCRIPTION	<b>Type de données</b> : VARCHAR(1024) ; <b>Null admis</b> : non  L'identificateur de message suivi par le texte de message. L'identificateur de message est le premier des sept caractères de la colonne DESCRIPTION. Le texte de message commence à la neuvième position de la colonne DESCRIPTION.

## Table ASN.IBMSNAP\_APPLYTRAIL

La table IBMSNAP\_APPLYTRAIL contient les informations d'analyse rétrospective de tous les cycles d'ensembles d'abonnements effectués par le programme Apply. Cette table enregistre un historique des mises à jour effectuées par rapport aux abonnements. Il s'agit d'un référentiel de diagnostic et de statistiques sur les performances. La table récapitulative Apply est l'un des meilleur endroit à consulter si un problème survient avec le programme Apply. Le programme Apply ne supprime pas automatiquement cette table, mais vous pouvez facilement rendre l'élagage automatique en ajoutant une instruction post SQL à l'un des ensembles d'abonnements.

**Serveur** : serveur de contrôle Apply

**Index** : LASTRUN, APPLY\_QUAL

Le tableau 90 fournit une brève description des colonnes dans la table IBMSNAP\_APPLYTRAIL.

Tableau 90. Colonnes dans la table IBMSNAP\_APPLYTRAIL

Nom de colonne	Description
APPLY_QUAL	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Identifie uniquement quel programme Apply a traité l'ensemble d'abonnements.
SET_NAME	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Nom de l'ensemble d'abonnements que le programme Apply a traité.
SET_TYPE	<b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non  Valeur qui apparaît dans la colonne SET_TYPE de la table IBMSNAP_SUBS_SET après le dernier cycle Apply.

Tableau 90. Colonnes dans la table IBMSNAP\_APPLYTRAIL (suite)

Nom de colonne	Description
WHOS_ON_FIRST	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Les valeurs suivantes sont utilisées pour contrôler l'ordre de traitement dans les scénarios de réplication bidirectionnelle.</p> <p><b>F</b> (premier) La table source est la réplique et la table cible est le maître. Dans le cas d'un conflit de mises à jour entre la réplique et la table maître, les transactions conflictuelles de la réplique seront rejetées. F n'est pas utilisé pour les abonnements en lecture seule ; il est utilisé pour les réplications bidirectionnelles.</p> <p><b>S</b> (premier) La table source est la table maître et la table cible est la réplique ou une autre copie. Dans le cas d'un conflit de mises à jour entre la table maître et la réplique, les transactions conflictuelles de la réplique seront rejetées. S est utilisé pour tous les abonnements en lecture seule.</p>
ASNLOAD	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Valeur utilisée pour démarrer le programme Apply :</p> <p><b>Y</b> Indique que le programme Apply a été démarré avec le paramètre <b>loadxit=y</b>, provoquant l'appel du sous-programme d'exit utilisateur ASNLOAD pour effectuer une régénération intégrale sur un ensemble d'abonnements.</p> <p><b>N</b> Indique que le programme d'exit n'a pas été appelé car une régénération intégrale n'était pas nécessaire ou le programme Apply n'a pas été démarré avec le paramètre <b>loadxit</b>.</p> <p><b>NULL</b> Indique qu'une erreur de programme Apply s'est produite avant que le programme Apply puisse déterminer si le programme d'exit ASNLOAD devait être appelé.</p>
FULL_REFRESH	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si une régénération intégrale s'est produite :</p> <p><b>Y</b> Indique qu'une régénération intégrale a été effectuée pour l'ensemble d'abonnements.</p> <p><b>N</b> Indique qu'aucune régénération intégrale n'a été effectuée pour l'ensemble d'abonnements.</p> <p><b>NULL</b> Indique qu'une erreur s'est produite avant que le programme Apply puisse déterminer si une régénération intégrale était nécessaire ou non.</p>
EFFECTIVE_MEMBERS	<p><b>Type de données</b> : INT ; <b>Null admis</b> : oui</p> <p>Nombre de membres d'un ensemble d'abonnements qui ont été modifiés pendant un cycle Apply, soit par régénération intégrale, soit par la réplication d'insertions, de mises à jour et de suppressions. Ce nombre est compris entre zéro et le nombre de membres d'un ensemble d'abonnements défini.</p>
SET_INSERTED	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre total de lignes insérées dans des membres d'un ensemble d'abonnements pendant le cycle d'abonnement.</p>
SET_DELETED	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre total de lignes supprimées des membres d'un ensemble d'abonnements pendant le cycle d'abonnement.</p>

Tableau 90. Colonnes dans la table IBMSNAP\_APPLYTRAIL (suite)

Nom de colonne	Description
SET_UPDATED	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre total de lignes mises à jour dans des membres d'un ensemble d'abonnements pendant le cycle d'abonnement.</p>
SET_REWORKED	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre total de lignes que le programme Apply transforme pendant le dernier cycle. Le programme Apply transforme des modifications dans les conditions suivantes :</p> <ul style="list-style-type: none"> <li>• Si une insertion échoue parce que la ligne existe déjà dans la table cible, le programme Apply convertit l'insertion en une mise à jour de la ligne existante.</li> <li>• Si une mise à jour échoue parce que la ligne n'existe pas dans la table cible, le programme Apply convertit la mise à jour en une insertion.</li> </ul>
SET_REJECTED_TRXS	<p><b>Type de données</b> : INT ; <b>Null admis</b> : non</p> <p>Nombre total de transactions rejetées en raison d'un conflit de répliquions bidirectionnelles. Cette colonne ne sert que pour les ensembles d'abonnements de répliquions bidirectionnelles où la détection de conflit est définie comme standard ou avancée.</p>

Tableau 90. Colonnes dans la table IBMSNAP\_APPLYTRAIL (suite)

Nom de colonne	Description
STATUS	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non</p> <p>Valeur représentant l'état de travail du programme Apply après un cycle donné :</p> <p><b>-1</b> La réplication a échoué. Le programme Apply a annulé la totalité des lignes qu'il avait appliquées et aucune donnée n'a été validée. Si le paramètre de démarrage SQLERRCONTINUE = Y, le SQLSTATE renvoyé au programme Apply pendant le dernier cycle ne fait <i>pas</i> partie des erreurs acceptables que vous avez indiquées dans le fichier en entrée pour SQLERRCONTINUE (<i>qualificatif Apply.SQS</i>).</p> <p><b>0</b> Le programme Apply a traité avec succès l'ensemble d'abonnements. Si le paramètre de démarrage SQLERRCONTINUE = Y, le programme Apply n'a rencontré aucune des erreurs SQL que vous avez indiquées pour le paramètre de démarrage SQLERRCONTINUE (dans <i>qualificatif Apply.SQS</i>) et n'a rejeté aucune ligne.</p> <p><b>2</b> Le programme Apply traite l'ensemble d'abonnements dans plusieurs cycles. Il a traité avec succès un abonnement logique simple divisé selon la colonne de contrôle MAX_SYNCH_MINUTES.</p> <p><b>16</b> Le programme Apply a traité avec succès l'ensemble d'abonnements et a renvoyé un état égal à 0 ; cependant, il a rencontré certaines des erreurs SQL que vous avez indiquées pour le paramètre de démarrage SQLERRCONTINUE (dans <i>qualificatif Apply.SQS</i>) et a rejeté certaines lignes. Pour des détails sur les lignes qui ont échoué, reportez-vous au fichier <i>qualificatif Apply.ERR</i>.</p> <p><b>Exemple</b> : vous avez défini SQLERRCONTINUE = Y et indiqué que l'état SQL autorisé est 23502 (code SQL -407). Une erreur 23502 s'est produite, mais aucune autre erreur ne s'est produite. Le programme Apply finit le traitement de l'ensemble d'abonnements et définit l'état à 16. Lors de l'exécution suivante, une erreur 23502 s'est produite, puis une erreur 07006 (code SQL -301). Maintenant le programme Apply arrête le traitement de l'ensemble d'abonnements, annule la totalité des lignes qu'il a appliquées et définit l'état à -1 (car aucune données n'a été validée).</p> <p><b>18</b> Le programme Apply traite l'ensemble d'abonnements dans plusieurs cycles et renvoie un état de 2, ce qui signifie qu'il a traité avec succès un abonnement logique simple divisé selon la colonne de contrôle MAX_SYNCH_MINUTES. Cependant, il a rencontré certaines des erreurs SQL que vous avez indiquées pour le paramètre de démarrage SQLERRCONTINUE (dans <i>qualificatif Apply.SQS</i>) et a rejeté certaines lignes. Pour des détails sur les lignes qui ont échoué, reportez-vous au fichier <i>qualificatif Apply.ERR</i>.</p>
LASTRUN	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non</p> <p>Heure approximative à laquelle le dernier abonnement a commencé. Le programme Apply définit la valeur LASTRUN chaque fois qu'un ensemble d'abonnements est traité. Il s'agit de l'heure approximative sur le serveur de contrôle à laquelle le programme Apply commence le traitement de l'ensemble d'abonnements.</p>
LASTSUCCESS	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : oui</p> <p>Horodatage du serveur de contrôle Apply pour le début du dernier traitement réussi d'un ensemble d'abonnements.</p>

Tableau 90. Colonnes dans la table IBMSNAP\_APPLYTRAIL (suite)

Nom de colonne	Description
SYNCHPOINT	<b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui  Le programme Apply utilise cette colonne pour enregistrer sa progression, indiquant qu'il a traité des données jusqu'à cette valeur de point de synchronisation pour l'ensemble d'abonnements.
SYNCHTIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : oui  Le programme Apply utilise cette colonne pour enregistrer sa progression, indiquant qu'il a traité des données jusqu'à cet horodatage pour l'ensemble d'abonnements.
SOURCE_SERVER	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Nom de la base de données DB2 où les tables et les vues sources sont définies.
SOURCE_ALIAS	<b>Type de données</b> : CHAR(8) ; <b>Null admis</b> : oui  Alias DB2 correspondant au serveur source nommé dans la colonne SOURCE_SERVER.
SOURCE_OWNER	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui  Qualificatif de haut niveau de la table ou de la vue source que le programme Apply a traitée. Cette valeur est définie uniquement lorsque le cycle Apply échoue.
SOURCE_TABLE	<b>Type de données</b> : VARCHAR(128), VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou précédente; <b>Null admis</b> : oui  Nom de la table ou de la vue source que le programme Apply a traitée. Cette valeur est définie uniquement lorsque le cycle Apply échoue.
SOURCE_VIEW_QUAL	<b>Type de données</b> : SMALLINT ; <b>Null admis</b> : oui  Valeur du qualificatif de vue source pour la table ou la vue source que le programme Apply a traité. Cette valeur est définie uniquement lorsque le cycle Apply échoue.
TARGET_SERVER	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Nom de la base de données du serveur où les tables ou les vues sources sont stockées.
TARGET_ALIAS	<b>Type de données</b> : CHAR(8) ; <b>Null admis</b> : oui  Alias DB2 correspondant au serveur cible nommé dans la colonne TARGET_SERVER.
TARGET_OWNER	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non  Qualificatif de haut niveau de la table cible que le programme Apply a traitée. Cette valeur est définie uniquement lorsque le cycle Apply échoue.
TARGET_TABLE	<b>Type de données</b> : VARCHAR(128), VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou précédente; <b>Null admis</b> : non  Nom de la table cible que le programme Apply a traitée. Cette valeur est définie uniquement lorsque le cycle Apply échoue.
CAPTURE_SCHEMA	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non  Nom de schéma des tables de serveur Capture pour cet ensemble d'abonnements.

Tableau 90. Colonnes dans la table IBMSNAP\_APPLYTRAIL (suite)

Nom de colonne	Description
TGT_CAPTURE_SCHEMA	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui</p> <p>Si la table cible est également la source d'un autre ensemble d'abonnements (comme par exemple une table CCD externe dans une configuration multiniveau ou une table réplique dans une configuration de réplication bidirectionnelle), cette colonne contient le schéma de Capture qui sera utilisé lorsque que la table sert de source.</p>
FEDERATED_SRC_SRVR	<p><b>Type de données</b> : VARCHAR(18) ; <b>Null admis</b> : oui</p> <p>Nom du serveur distant fédéré qui est la source de l'ensemble d'abonnements qui s'applique uniquement aux sources relationnelles non-DB2.</p>
FEDERATED_TGT_SRVR	<p><b>Type de données</b> : VARCHAR(18) ; <b>Null admis</b> : oui</p> <p>Nom du serveur distant fédéré qui est la cible de l'ensemble d'abonnements qui s'applique uniquement aux serveurs cibles relationnels non-DB2.</p>
JRN_LIB	<p><b>Type de données</b> : CHAR(10) ; <b>Null admis</b> : oui</p> <p><b>System i</b> Cette colonne, qui s'applique uniquement aux serveurs Capture System i, est le nom de bibliothèque du journal que la table source utilise.</p>
JRN_NAME	<p><b>Type de données</b> : CHAR(10) ; <b>Null admis</b> : oui</p> <p><b>System i</b> Cette colonne, qui s'applique uniquement aux serveurs Capture System i, est le nom du journal utilisé par la table source. Dans cette colonne, un astérisque suivi de neuf blancs signifie qu'actuellement la table source n'est pas dans un journal ; dans ce cas, il n'est pas possible de capturer des données pour cette table source.</p>
COMMIT_COUNT	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : oui</p> <p>Valeur de COMMIT_COUNT du dernier cycle du programme Apply enregistré dans la table IBMSNAP_SUBS_SET.</p>
OPTION_FLAGS	<p><b>Type de données</b> : CHAR(4) ; <b>Null admis</b> : non</p> <p>Réservé aux options ultérieures de la réplication SQL. Actuellement, cette colonne contient la valeur par défaut de NNNN.</p>
EVENT_NAME	<p><b>Type de données</b> : CHAR(18) ; <b>Valeur NULL admise</b> : oui</p> <p>Chaîne de caractères unique utilisée pour représenter l'événement qui a déclenché le traitement de l'ensemble.</p>
ENDTIME	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non, avec valeur par défaut ; <b>Défaut</b> : horodatage actuel.</p> <p>Horodatage sur le serveur de contrôle Apply lorsque le programme Apply a fini le traitement de l'ensemble d'abonnements. Pour savoir combien de temps a duré un traitement, soustrayez LASTRUN de ENDTIME.</p>
SOURCE_CONN_TIME	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : oui</p> <p>Horodatage sur le serveur de contrôle Capture lorsque le programme Apply s'est connecté pour la première fois pour extraire la données source.</p>
SQLSTATE	<p><b>Type de données</b> : CHAR(5) ; <b>Null admis</b> : oui</p> <p>Code d'état SQL pour une exécution ayant échoué. Sinon, NULL.</p>

Tableau 90. Colonnes dans la table IBMSNAP\_APPLYTRAIL (suite)

Nom de colonne	Description
SQLCODE	<b>Type de données</b> : INT ; <b>Null admis</b> : oui  Code d'erreur SQL pour une exécution ayant échoué. Sinon, NULL.
SQLERRP	<b>Type de données</b> : CHAR(8) ; <b>Null admis</b> : oui  Identificateur du produit de base de données du serveur où une erreur SQL s'est produite, entraînant l'échec de l'exécution. Sinon, NULL.
SQLERRM	<b>Type de données</b> : VARCHAR(70) ; <b>Null admis</b> : oui  Information d'erreur SQL pour une exécution ayant échoué.
APPERRM	<b>Type de données</b> : VARCHAR(760) ; <b>Null admis</b> : oui  ID et texte de message d'erreur Apply pour une exécution ayant échoué.

## Table ASN.IBMSNAP\_FEEDETL

La table ASN.IBMSNAP\_FEEDETL identifie les membres d'un ensemble d'abonnements de réplication SQL utilisables par InfoSphere DataStage afin d'alimenter un entrepôt de données. La table stocke également des informations de point de synchronisation liées à DataStage pour les membres d'un ensemble d'abonnements et destinées à suivre la progression de DataStage.

**Serveur** : serveur de contrôle Apply

**Clé primaire** : APPLY\_QUAL, SET\_NAME, SOURCE\_OWNER, SOURCE\_TABLE, TARGET\_OWNER, TARGET\_TABLE

**Important** : Prenez garde lorsque vous mettez à jour cette table avec 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 de la table IBMSNAP\_FEEDETL.

Tableau 91. Colonnes dans la table IBMSNAP\_FEEDETL

Nom de colonne	Description
APPLY_QUAL	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Identifie uniquement quel programme Apply traite ce membre d'un ensemble d'abonnements.
SET_NAME	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Nom de l'ensemble d'abonnements auquel ce membre appartient.
SOURCE_OWNER	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non  Qualificatif de haut niveau pour la table ou la vue source pour ce membre.
SOURCE_TABLE	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non  Nom de la table ou de la vue source pour ce membre.
TARGET_OWNER	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non  Qualificatif de haut niveau pour la table ou la vue cible pour ce membre.



Tableau 91. Colonnes dans la table IBMSNAP\_FEEDETL (suite)

Nom de colonne	Description
TARGET_TABLE	<b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non  Nom de la table ou de la vue cible pour ce membre.
MIN_SYNCHPOINT	<b>Type de données</b> : CHAR(10) FOR BIT DATA ; <b>Nul admis</b> : non  Point de départ d'un intervalle de lignes extraites d'un membre de table CDD et alimentant DataStage. Lors de la première exécution du travail d'extraction, la valeur est x'00000000000000000000'. A l'issue d'un cycle DataStage réussi, cette colonne est mise à jour avec la valeur MAX_SYNCHPOINT de la ligne.
MAX_SYNCHPOINT	<b>Type de données</b> : CHAR(10) FOR BIT DATA ; <b>Nul admis</b> : non  Point d'extrémité d'un intervalle de lignes extraites d'un membre de table CDD et alimentant DataStage. Au début du cycle, cette colonne est mise à jour avec la valeur de la colonne SYNCHPOINT de la table IBMSNAP_SUBSET pour l'ensemble d'abonnements contenant le membre de table CDD. A la fin du cycle, cette valeur est insérée dans la colonne MIN_SYNCHPOINT et devient le point de départ du prochain cycle DataStage.
DSX_CREATE_TIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non  Heure de création du nouveau fichier de définition DataStage (.dsx).
DSX_UPDATE_TIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non  Heure de mise à jour du fichier de définition DataStage (.dsx).

## Table ASN.IBMSNAP\_SUBS\_COLS

La table IBMSNAP\_SUBS\_COLS contient des informations sur les colonnes des membres d'ensemble d'abonnements copiés dans un ensemble d'abonnements. Des lignes sont automatiquement insérées dans cette table ou supprimées de cette table lorsque des informations sont modifiées dans une ou plusieurs colonne d'une paire de tables source et cible. Utilisez cette table si vous avez besoin d'informations sur des colonnes spécifiques dans un membre d'un ensemble d'abonnements.

**Serveur** : serveur de contrôle Apply

**Index** : APPLY\_QUAL, SET\_NAME, WHOS\_ON\_FIRST, TARGET\_OWNER, TARGET\_TABLE, TARGET\_NAME

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 92 fournit une brève description des colonnes dans la table IBMSNAP\_SUBS\_COLS.

Tableau 92. Colonnes dans la table IBMSNAP\_SUBS\_COLS

Nom de colonne	Description
APPLY_QUAL	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Identifie uniquement quel programme Apply traite ce membre d'un ensemble d'abonnements.
SET_NAME	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Nom d'un ensemble d'abonnements auquel ce membre appartient.

Tableau 92. Colonnes dans la table IBMSNAP\_SUBS\_COLS (suite)

Nom de colonne	Description
WHOS_ON_FIRST	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Les valeurs suivantes sont utilisées pour contrôler l'ordre de traitement dans les scénarios de réplication bidirectionnelle.</p> <p><b>F</b> (premier) La table source est la réplique et la table cible est le maître. Dans le cas d'un conflit de mises à jour entre la réplique et la table maître, les transactions conflictuelles de la réplique seront rejetées. F n'est pas utilisé pour les abonnements en lecture seule ; il est utilisé pour les réplifications bidirectionnelles.</p> <p><b>S</b> (premier) La table source est la table maître et la table cible est la réplique ou une autre copie. Dans le cas d'un conflit de mises à jour entre la table maître et la réplique, les transactions conflictuelles de la réplique seront rejetées. S est utilisé pour tous les abonnements en lecture seule.</p>
TARGET_OWNER	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non</p> <p>Qualificatif de haut niveau pour une table ou une vue cible.</p>
TARGET_TABLE	<p><b>Type de données</b> : VARCHAR(128) ; VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou précédente; <b>Null admis</b> : non</p> <p>Table ou vue à laquelle les données sont appliquées.</p>
COL_TYPE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Indicateur signalant le type de colonne :</p> <p><b>A</b> Colonne image après.</p> <p><b>B</b> Colonne image avant.</p> <p><b>C</b> Colonne calculée ou expression SQL utilisant des fonctions scalaires.</p> <p><b>F</b> Colonne calculée utilisant des fonctions de colonne.</p> <p><b>L</b> Valeur d'indicateur LOB.</p> <p><b>P</b> Colonne de prédicat image avant.</p> <p><b>R</b> Colonne de numéro relatif d'enregistrement fournie par le système et utilisée comme colonne de clé primaire. Utilisée uniquement par DB2 DataPropagator pour System i.</p>
TARGET_NAME	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non</p> <p>Nom de la colonne de table ou de vue cible. Il n'est pas nécessaire qu'il corresponde au nom de colonne source.</p> <p>Les noms de colonnes CCD internes ne peuvent pas être modifiés. Ils doivent correspondre aux noms de colonnes de tables sources.</p>
IS_KEY	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Indicateur signalant si la colonne fait partie de la clé cible qui peut être un index à entrées uniques ou une clé primaire de table cible condensée :</p> <p><b>Y</b> La colonne fait partie de la clé cible.</p> <p><b>N</b> La colonne ne fait pas partie de la clé cible.</p>

Tableau 92. Colonnes dans la table IBMSNAP\_SUBS\_COLS (suite)

Nom de colonne	Description
COLNO	<b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non  Emplacement numérique de la colonne dans la source originale source, à conserver par rapport aux autres colonnes utilisateur dans les affichages et les abonnements.
EXPRESSION	<b>Type de données</b> : VARCHAR(1024) ; <b>Null admis</b> : non  Nom de la colonne source ou expression SQL utilisée pour créer le contenu de la colonne cible.

## Table ASN.IBMSNAP\_SUBS\_EVENT

La table IBMSNAP\_SUBS\_EVENT contient des informations relatives aux déclencheurs d'événements associés à un ensemble d'abonnements. Elle contient également des noms et des horodatages associés aux noms d'événement.

**Serveur** : serveur de contrôle Apply

**Index** : EVENT\_NAME, EVENT\_TIME

Cette table contient des informations que vous pouvez mettre à jour à l'aide de SQL.

Vous insérez une ligne dans cette table lorsque vous créez un nouvel événement pour démarrer un programme Apply.

Le tableau 93 fournit une brève description des colonnes dans la table IBMSNAP\_SUBS\_EVENT.

Tableau 93. Colonnes dans la table IBMSNAP\_SUBS\_EVENT

Nom de colonne	Description
EVENT_NAME	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Identificateur unique d'un événement. Cet identificateur sert à déclencher la réplication pour un ensemble d'abonnements.
EVENT_TIME	<b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non  Horodatage du serveur de contrôle Apply d'une heure de message actuel ou futur. Les applications utilisateurs signalant les événements de réplication fournissent les valeurs de cette colonne.
END_SYNCHPOINT	<b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui  Numéro LSN indiquant au programme Apply d'appliquer uniquement les données qui ont été capturées jusque là. Vous pouvez trouver la valeur END_SYNCHPOINT exacte que vous souhaitez utiliser en vous reportant à la table de signaux et en trouvant le numéro LSN associé à un horodatage. Les transactions validées après ce point dans le journal ne sont pas répliquées tant qu'aucun événement ultérieur n'est posté. Si vous fournissez des valeurs pour END_SYNCHPOINT et END_OF_PERIOD, le programme Apply utilise la valeur END_SYNCHPOINT car ainsi il n'a pas besoin d'effectuer de calculs à partir des table de contrôle pour trouver le numéro LSN maximum à répliquer.

Tableau 93. Colonnes dans la table IBMSNAP\_SUBS\_EVENT (suite)

Nom de colonne	Description
END_OF_PERIOD	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : oui</p> <p>Horodatage utilisé par le programme Apply qui applique uniquement les données qui ont été journalisées jusque là. Les transactions validées après ce point dans le journal ne sont pas répliquées tant qu'aucun événement ultérieur n'est posté.</p>

## Table ASN.IBMSNAP\_SUBS\_MEMBR

La table IBMSNAP\_SUBS\_MEMBR contient des informations relatives aux paires de tables source et cible individuelles définies pour un ensemble d'abonnements. Une seule ligne est automatiquement insérée dans cette table lorsque vous ajoutez un membre d'ensemble d'abonnements. Utilisez cette table pour identifier une paire de tables source et cible spécifique dans un ensemble d'abonnements.

**Serveur** : serveur de contrôle Apply

**Index** : APPLY\_QUAL, SET\_NAME, WHOS\_ON\_FIRST, SOURCE\_OWNER, SOURCE\_TARGET, SOURCE\_VIEW\_QUAL, TARGET\_OWNER, TARGET\_TABLE

**Important** : Prenez garde au moment de mettre à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 94 fournit une brève description des colonnes dans la table IBMSNAP\_SUBS\_MEMBR.

Tableau 94. Colonnes dans la table IBMSNAP\_SUBS\_MEMBR

Nom de colonne	Description
APPLY_QUAL	<p><b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non</p> <p>Identifie uniquement quel programme Apply traite ce membre d'un ensemble d'abonnements.</p>
SET_NAME	<p><b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non</p> <p>Nom de l'ensemble d'abonnements auquel ce membre appartient.</p>
WHOS_ON_FIRST	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Les valeurs suivantes sont utilisées pour contrôler l'ordre de traitement dans les scénarios de réplification bidirectionnelle.</p> <p><b>F</b> (premier) La table source est la réplique et la table cible est le maître. Dans le cas d'un conflit de mises à jour entre la réplique et la table maître, les transactions conflictuelles de la réplique seront rejetées. F n'est pas utilisé pour les abonnements en lecture seule ; il est utilisé pour les réplifications bidirectionnelles.</p> <p><b>S</b> (premier) La table source est la table maître et la table cible est la réplique ou une autre copie. Dans le cas d'un conflit de mises à jour entre la table maître et la réplique, les transactions conflictuelles de la réplique seront rejetées. S est utilisé pour tous les abonnements en lecture seule.</p>
SOURCE_OWNER	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non</p> <p>Qualificatif de haut niveau pour la table ou la vue source pour ce membre.</p>

Tableau 94. Colonnes dans la table IBMSNAP\_SUBS\_MEMBR (suite)

Nom de colonne	Description
SOURCE_TABLE	<p><b>Type de données</b> : VARCHAR(128) ; VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou précédente ; <b>Null admis</b> : non</p> <p>Nom de la table ou de la vue source pour ce membre.</p>
SOURCE_VIEW_QUAL	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non</p> <p>Prend en charge la vue des tables physiques par correspondance avec la colonne similaire de la table IBMSNAP_REGISTER. Cette valeur est égale à 0 pour les tables physiques définies en tant que sources et est supérieure à 0 pour les vues définies en tant que sources. Cette colonne sert à prendre en charge plusieurs abonnements pour différentes vues sources avec des valeurs de colonnes SOURCE_OWNER et SOURCE_TABLE identiques.</p>
TARGET_OWNER	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non</p> <p>Qualificatif de haut niveau pour la table ou la vue cible pour ce membre.</p>
TARGET_TABLE	<p><b>Type de données</b> : VARCHAR(128), VARCHAR(18) pour DB2 UDB pour les sous-systèmes de mode compatibilité z/OS version 8 ou précédente ; <b>Null admis</b> : non</p> <p>Nom de la table ou de la vue cible pour ce membre.</p>
TARGET_CONDENSED	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Indicateur signalant :</p> <p><b>Y</b> Pour une valeur de clé primaire donnée, la table cible affiche une seule ligne.</p> <p><b>N</b> Toutes les modifications doivent rester afin de conserver un historique de mises à jour complet.</p> <p><b>A</b> La table cible est une table d'agrégation de base ou une table d'agrégation des modifications.</p>
TARGET_COMPLETE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Indicateur signalant :</p> <p><b>Y</b> La table cible contient une ligne par valeur de clé primaire présentant un intérêt.</p> <p><b>N</b> La table cible contient un sous-ensemble de lignes de valeurs de clé primaire.</p>
TARGET_STRUCTURE	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non</p> <p>Structure de la table cible :</p> <p><b>1</b> Table utilisateur</p> <p><b>3</b> Table de modification cohérente des données</p> <p><b>4</b> Table des points de cohérence</p> <p><b>5</b> Table d'agrégation de base</p> <p><b>6</b> Table d'agrégation des modifications</p> <p><b>7</b> Réplique</p> <p><b>8</b> Copie utilisateur</p> <p><b>9</b> Table de modification cohérente des données sans jointure des tables IBMSNAP_UOW et CD</p>

Tableau 94. Colonnes dans la table IBMSNAP\_SUBS\_MEMBR (suite)

Nom de colonne	Description
PREDICATES	<p><b>Type de données</b> : VARCHAR(1024) ; <b>Null admis</b> : oui</p> <p>Liste les prédicats à placer dans une clause WHERE pour la table dans la colonne TARGET_TABLE. Cette clause WHERE crée un sous-ensemble de lignes de la table source. Les prédicats sont reconnus uniquement lorsque WHOS_ON_FIRST est égal à S. Le prédicat ne peut pas contenir de clause ORDER BY car le programme Apply ne peut pas générer de clause ORDER BY. Les tables d'agrégation nécessitent un prédicat factice suivi d'une clause GROUP BY.</p> <p>Le programme Apply utilisant ces prédicats à la fois pour la régénération intégrale et pour la réplication en mode capture des modifications, cette colonne ne peut pas contenir de prédicats impliquant des colonnes de la table de modification des données ou de la table UOW. Les prédicats contenant des références à la table de modification des données ou à la table UOW sont stockés dans la colonne UOW_CD_PREDICATES.</p>
MEMBER_STATE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant dans quel état est le membre :</p> <p><b>N</b> (Nouveau) Le membre est nouveau dans cet ensemble d'abonnements. Tous les membres récemment activés apparaîtront donc dans cet état.</p> <p><b>L</b> (Chargé) Les membres de cet ensemble d'abonnements ont été chargés, mais aucun cycle de capture des modifications n'a encore eu lieu.</p> <p><b>S</b> (Synchronisé) Le membre a été avancé de l'état nouveau (N) à l'état chargé (L) et est maintenant synchronisé avec tous les autres membres de l'ensemble d'abonnements qui sont dans l'état synchronisé. Lorsque tous les membres d'un ensemble d'abonnements sont dans l'état synchronisé, des réplication de modification peuvent se produire au niveau de l'ensemble d'abonnements.</p> <p><b>D</b> (Désactivé) Le membre est désactivé pour cet ensemble d'abonnements.</p>
TARGET_KEY_CHG	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Indicateur signalant comment le programme Apply gère les mises à jour lorsque, sur la table source, vous remplacez les colonnes sources par les colonnes de clés cibles d'une table cible :</p> <p><b>Y</b> Le programme Apply met à jour la table cible en fonction des images avant de la colonne de clés cibles, ce qui signifie que le programme Apply modifie le prédicat avec les anciennes valeurs au lieu des nouvelles. Assurez-vous que vous avez enregistré toutes les colonnes d'images avant de la clé cible afin qu'elles soient présentes dans la table de modification des données. Pour l'entrée d'enregistrement correspondant dans la table de registres, assurez-vous que la valeur dans la colonne CHG_UPD_TO_DEL_INS est égale à N.</p> <p><b>N</b> Le programme Apply utilise une logique lors du traitement des mises à jour et des suppressions qui supposent que les colonnes composant la clé cible ne sont jamais mises à jour.</p>

Tableau 94. Colonnes dans la table IBMSNAP\_SUBS\_MEMBR (suite)

Nom de colonne	Description
UOW_CD_PREDICATES	<p><b>Type de données</b> : VARCHAR(1024) ; <b>Null admis</b> : oui</p> <p>Contient des prédicats incluant des colonnes de la table de modification des données ou de la table UOW dont le programme Apply a besoin uniquement pour la réplication en mode capture des modifications et non pour les régénérations intégrales. Lors de la réplication en mode capture des modifications, le programme Apply traite les prédicats de cette colonne et de la colonne PREDICATES. Lors de la régénération intégrale, le programme Apply traite uniquement les prédicats de la colonne PREDICATES.</p>
JOIN_UOW_CD	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : oui</p> <p>Indicateur signalant si le programme Apply effectue une jointure des tables de modification des données et UOW lors du traitement d'une table cible de copie utilisateur. Cet indicateur est nécessaire lorsque vous définissez un membre d'un ensemble d'abonnements avec des prédicats utilisant des colonnes de la table UOW qui ne sont pas dans la table de modification des données. Si le type de table cible est différent de la copie utilisateur, le programme Apply utilise une jointure des tables de modification des données et UOW lors du traitement du membre et ignore cette colonne lors du traitement du membre.</p> <p><b>Y</b> Le programme Apply utilise une jointure des tables de modification des données et UOW lors du traitement du membre.</p> <p><b>N</b> Le programme Apply n'utilise pas de jointure des tables de modification des données et UOW lors du traitement du membre ; il lit les modifications uniquement dans la table de modification des données.</p> <p><b>NULL</b> Le programme Apply ignore cette colonne lors du traitement du membre. Si la table cible est une copie utilisateur et que la valeur dans cette colonne est Null, le programme Apply n'utilise pas de jointure des tables de modification des données et UOW lors du traitement du membre.</p>

Tableau 94. Colonnes dans la table IBMSNAP\_SUBS\_MEMBR (suite)

Nom de colonne	Description
LOADX_TYPE	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : oui</p> <p>Type de chargement pour ce membre. La valeur dans cette colonne est utilisée pour contourner les valeurs par défaut. Cette valeur n'est utilisable que lorsque le paramètre d'appel Apply <b>loadxit=y</b> est indiqué.</p> <p>NULL</p> <p><b>z/OS</b> La fonction LOAD from CURSOR est utilisée pour ce membre.</p> <p><b>Linux UNIX Windows</b> La routine d'exit ASNLOAD détermine l'utilitaire le plus approprié pour ce membre (option 3, 4 ou 5).</p> <p>1 ASNLOAD n'est pas utilisé pour ce membre. Ceci désactive l'option ASNLOAD pour un membre d'un ensemble d'abonnements particulier même si vous avez spécifié LOADX au démarrage.</p> <p>2 Un code de sortie ASNLOAD défini ou modifié par l'utilisateur est utilisé.</p> <p>3 La fonction LOAD from CURSOR est utilisée pour ce membre.</p> <p><b>Linux UNIX Windows</b> 4 EXPORT est utilisé pour ce membre.</p> <p><b>Linux UNIX Windows</b> 5 EXPORT et IMPORT sont utilisés pour ce membre.</p> <p>6 La table cible ne sera pas chargée pour ce membre.</p> <p><b>Restriction</b> : <b>Linux UNIX Windows</b> L'utilitaire LOAD n'est pas pris en charge pour les tables classifiées. Pour effectuer une régénération intégrale d'une table classifiée, vous pouvez utiliser l'utilitaire DB2 IMPORT ou le programme Apply pour faire une régénération intégrale de la table par l'intermédiaire de SQL.</p>
LOADX_SRC_N_OWNER	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui</p> <p>Propriétaire de pseudonyme créé par l'utilisateur. Cette valeur est nécessaire lorsque toutes les conditions suivantes sont réunies :</p> <ul style="list-style-type: none"> <li>• La fonction LOAD from CURSOR est utilisée pour ce membre (LOADX_TYPE est égal à 3)</li> <li>• Le serveur cible est Linux, UNIX, or Windows</li> <li>• La source n'est pas un pseudonyme</li> </ul>
LOADX_SRC_N_TABLE	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui</p> <p>Table de pseudonyme créé par l'utilisateur. Cette valeur est nécessaire lorsque toutes les conditions suivantes sont réunies :</p> <ul style="list-style-type: none"> <li>• La fonction LOAD from CURSOR est utilisée pour ce membre (LOADX_TYPE est égal à 3)</li> <li>• Le serveur cible est Linux, UNIX, or Windows</li> <li>• La source n'est pas un pseudonyme</li> </ul>

## Table ASN.IBMSNAP\_SUBS\_SET

La table IBMSNAP\_SUBS\_SET liste tous les ensembles d'abonnements définis sur le serveur de contrôle Apply et documente la progression de la réplication pour ces ensembles. Des lignes sont insérées dans cette table lorsque vous créez la définition de votre ensemble d'abonnements.



**Serveur** : serveur de contrôle Apply

**Index** : APPLY\_QUAL, SET\_NAME, WHOS\_ON\_FIRST

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données.

Le tableau 95 fournit une brève description des colonnes dans la table IBMSNAP\_SUBS\_SET.

Tableau 95. Colonnes dans la table IBMSNAP\_SUBS\_SET

Nom de colonne	Description
APPLY_QUAL	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non Identifie uniquement quel programme Apply traite cet ensemble d'abonnements.
SET_NAME	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non Nom de l'ensemble d'abonnements.
SET_TYPE	<b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non Indicateur signalant si l'ensemble est en lecture seule ou en lecture-écriture : <b>R</b> L'ensemble est en lecture seule. <b>U</b> L'ensemble est une configuration bidirectionnelle et est donc en lecture-écriture.
WHOS_ON_FIRST	<b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non Les valeurs suivantes sont utilisées pour contrôler l'ordre de traitement dans les scénarios de réplication bidirectionnelle. <b>F</b> (premier) La table source est la réplique et la table cible est le maître. Dans le cas d'un conflit de mises à jour entre la réplique et la table maître, les transactions conflictuelles de la réplique seront rejetées. F n'est pas utilisé pour les abonnements en lecture seule ; il est utilisé pour les réplifications bidirectionnelles. <b>S</b> (premier) La table source est la table maître et la table cible est la réplique ou une autre copie. Dans le cas d'un conflit de mises à jour entre la table maître et la réplique, les transactions conflictuelles de la réplique seront rejetées. S est utilisé pour tous les abonnements en lecture seule.
ACTIVATE	<b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non Indicateur signalant si le programme Apply traitera l'ensemble lors de son prochain cycle : <b>0</b> L'ensemble d'abonnements est désactivé. Le programme Apply ne traitera pas l'ensemble. <b>1</b> L'ensemble d'abonnements est actif indéfiniment. Le programme Apply traitera l'ensemble lors de chaque cycle du programme Apply jusqu'à ce que vous désactiviez l'ensemble ou jusqu'à ce que le programme Apply soit dans l'incapacité de le traiter. <b>2</b> L'ensemble d'abonnements est actif pour un seul cycle du programme Apply. Le programme Apply traitera l'ensemble une fois puis désactivera l'ensemble.

Tableau 95. Colonnes dans la table IBMSNAP\_SUBS\_SET (suite)

Nom de colonne	Description
SOURCE_SERVER	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Nom de la base de données du serveur de contrôle de Capture où les tables et les vues sources sont définies.
SOURCE_ALIAS	<b>Type de données</b> : CHAR(8) ; <b>Null admis</b> : oui  Alias DB2 correspondant au serveur de contrôle de Capture nommé dans la colonne SOURCE_SERVER.
TARGET_SERVER	<b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non  Nom de la base de données du serveur où les tables ou les vues sources sont stockées.
TARGET_ALIAS	<b>Type de données</b> : CHAR(8) ; <b>Null admis</b> : oui  Alias DB2 correspondant au serveur cible nommé dans la colonne TARGET_SERVER.

Tableau 95. Colonnes dans la table IBMSNAP\_SUBS\_SET (suite)

Nom de colonne	Description
STATUS	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non</p> <p>Valeur représentant l'état de travail du programme Apply après un cycle donné :</p> <p><b>-1</b> La réplication a échoué. Le programme Apply a annulé la totalité des lignes qu'il avait appliquées et aucune donnée n'a été validée. Si le paramètre de démarrage SQLERRCONTINUE = Y, le SQLSTATE renvoyé au programme Apply pendant le dernier cycle ne fait pas partie des erreurs acceptables que vous avez indiquées dans le fichier en entrée pour SQLERRCONTINUE (<i>qualificatif Apply.SQS</i>).</p> <p><b>0</b> Le programme Apply a traité avec succès l'ensemble d'abonnements. Si le paramètre de démarrage SQLERRCONTINUE = Y, le programme Apply n'a rencontré aucune des erreurs SQL que vous avez indiquées pour le paramètre de démarrage (dans <i>qualificatif Apply.SQS</i>) et n'a rejeté aucune ligne.</p> <p><b>1</b> Le programme Apply traite l'ensemble d'abonnements.</p> <p><b>2</b> Le programme Apply traite l'ensemble d'abonnements dans plusieurs cycles. Il a traité avec succès un abonnement logique simple divisé selon la colonne de contrôle MAX_SYNCH_MINUTES.</p> <p><b>16</b> Le programme Apply a traité avec succès l'ensemble d'abonnements et a renvoyé un état égal à 0 ; cependant, il a rencontré certaines des erreurs SQL que vous avez indiquées pour le paramètre de démarrage SQLERRCONTINUE (dans <i>qualificatif Apply.SQS</i>) et a rejeté certaines lignes. Pour des détails sur les lignes qui ont échoué, reportez-vous au fichier <i>qualificatif Apply.ERR</i>.</p> <p><b>Exemple</b> : vous avez défini SQLERRCONTINUE = Y et indiqué que l'état SQL autorisé est 23502 (code SQL -407). Une erreur 23502 s'est produite, mais aucune autre erreur ne s'est produite. Le programme Apply finit le traitement de l'ensemble d'abonnements et définit l'état à 16. Lors de l'exécution suivante, une erreur 23502 s'est produite, puis une erreur 07006 (code SQL -301). Maintenant le programme Apply arrête le traitement de l'ensemble d'abonnements, annule la totalité des lignes qu'il a appliquées et définit l'état à -1 (car aucune données n'a été validée).</p> <p><b>18</b> Le programme Apply traite l'ensemble d'abonnements dans plusieurs cycles et renvoie un état de 2, ce qui signifie qu'il a traité avec succès un abonnement logique simple divisé selon la colonne de contrôle MAX_SYNCH_MINUTES. Cependant, il a rencontré certaines des erreurs SQL que vous avez indiquées pour le paramètre de démarrage SQLERRCONTINUE (dans <i>qualificatif Apply.SQS</i>) et a rejeté certaines lignes. Pour des détails sur les lignes qui ont échoué, reportez-vous au fichier <i>qualificatif Apply.ERR</i>.</p>
LASTRUN	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : non</p> <p>Heure approximative à laquelle le dernier ensemble d'abonnements a commencé. Le programme Apply définit la valeur LASTRUN chaque fois qu'un ensemble d'abonnements est traité. Il s'agit de l'heure approximative sur le serveur de contrôle à laquelle le programme Apply commence le traitement de l'ensemble d'abonnements.</p>

Tableau 95. Colonnes dans la table IBMSNAP\_SUBS\_SET (suite)

Nom de colonne	Description
REFRESH_TYPE	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Type de planification utilisée pour demander au programme Apply de traiter cet ensemble d'abonnements :</p> <p><b>R</b> Le programme Apply utilise une planification basée sur le temps. Il utilise la valeur dans SLEEP_MINUTES pour déterminer le moment où commencer le traitement de l'ensemble d'abonnements.</p> <p><b>E</b> Le programme Apply utilise une planification basée sur les événements. Il contrôle la valeur de temps dans la table IBMSNAP_SUBS_EVENT pour déterminer le moment où commencer le traitement de l'ensemble d'abonnements. Avant qu'une réplication (capture de modification ou régénération intégrale) puisse commencer, un événement doit se produire.</p> <p><b>B</b> Le programme Apply utilise une planification basée à la fois sur le temps et sur les événements. Il traite alors l'ensemble d'abonnements suivant un critère de temps ou d'événement.</p>
SLEEP_MINUTES	<p><b>Type de données</b> : INT ; <b>Null admis</b> : oui</p> <p>Définit la durée (en minutes) d'inactivité entre les traitements d'ensemble d'abonnements. Le temps de traitement n'est utilisé que lorsque REFRESH_TYPE est égal à R ou B. Si la valeur de SLEEP_MINUTES est NULL, le programme Apply traitera l'ensemble en continu. Le programme Apply traitera l'ensemble aussi souvent que possible, mais traitera également tous les autres ensembles d'abonnements actifs ayant le même qualificatif Apply.</p>
EVENT_NAME	<p><b>Type de données</b> : CHAR(18) ; <b>Valeur NULL admise</b> : oui</p> <p>Chaîne de caractères unique utilisée pour représenter le nom d'événement. Utilisez cet identificateur pour mettre à jour les tables d'événements d'abonnement lorsque vous voulez déclencher la réplication pour un ensemble d'abonnements. Le nom d'événement n'est utilisé que lorsque REFRESH_TYPE est égal à E ou B.</p>
LASTSUCCESS	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : oui</p> <p>Horodatage du serveur de contrôle Apply pour le début du dernier traitement réussi d'un ensemble d'abonnements.</p>
SYNCHPOINT	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui</p> <p>Le programme Apply utilise cette colonne pour enregistrer sa progression, indiquant qu'il a traité des données jusqu'à cette valeur de point de synchronisation pour l'ensemble d'abonnements.</p>
SYNCHTIME	<p><b>Type de données</b> : TIMESTAMP ; <b>Null admis</b> : oui</p> <p>Le programme Apply utilise cette colonne pour enregistrer sa progression, indiquant qu'il a traité des données jusqu'à cet horodatage pour l'ensemble d'abonnements.</p>
CAPTURE_SCHEMA	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : non</p> <p>Nom de schéma des tables de contrôle Capture traitant la source pour cet ensemble d'abonnements.</p>

Tableau 95. Colonnes dans la table IBMSNAP\_SUBS\_SET (suite)

Nom de colonne	Description
TGT_CAPTURE_SCHEMA	<p><b>Type de données</b> : VARCHAR(128) ; <b>Null admis</b> : oui</p> <p>Si la table cible est également la source d'un autre ensemble d'abonnements (comme par exemple une table CCD externe dans une configuration multiniveau ou une table réplique dans une configuration de réplication bidirectionnelle), cette colonne contient le schéma de Capture qui est utilisé lorsque que la table sert de source.</p>
FEDERATED_SRC_SRVR	<p><b>Type de données</b> : VARCHAR(18) ; <b>Null admis</b> : oui</p> <p>Nom du serveur distant fédéré qui est la source de l'ensemble d'abonnements qui s'applique uniquement aux sources relationnelles non-DB2.</p>
FEDERATED_TGT_SRVR	<p><b>Type de données</b> : VARCHAR(18) ; <b>Null admis</b> : oui</p> <p>Nom du serveur distant fédéré qui est la cible de l'ensemble d'abonnements qui s'applique uniquement aux cibles relationnelles non-DB2.</p>
JRN_LIB	<p><b>Type de données</b> : CHAR(10) ; <b>Null admis</b> : oui</p> <p><b>System i</b> Cette colonne, qui s'applique uniquement aux serveurs Capture System i, est le nom de bibliothèque du journal que la table source utilise.</p>
JRN_NAME	<p><b>Type de données</b> : CHAR(10) ; <b>Null admis</b> : oui</p> <p><b>System i</b> Cette colonne, qui s'applique uniquement aux serveurs Capture System i, est le nom du journal utilisé par la table source. Dans cette colonne, un astérisque suivi de neuf blancs signifie qu'actuellement la table source n'est pas dans un journal ; dans ce cas, il n'est pas possible de capturer des données pour cette table source.</p>
OPTION_FLAGS	<p><b>Type de données</b> : CHAR(4) ; <b>Null admis</b> : non</p> <p>Réservé aux options ultérieures de la réplication SQL. Actuellement, cette colonne contient la valeur par défaut de NNNN.</p>

Tableau 95. Colonnes dans la table IBMSNAP\_SUBS\_SET (suite)

Nom de colonne	Description
COMMIT_COUNT	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : oui</p> <p>Indicateur signalant le type de traitement que le programme Apply effectue pour un ensemble d'abonnements :</p> <p><b>NULL</b> Il s'agit de la valeur par défaut pour un ensemble d'abonnements en lecture seule. Le programme Apply traitera les ensembles de réponses extraits pour les membres d'un ensemble d'abonnements <i>n</i> un par un jusqu'à ce que toutes les données soient traitées, puis émettra une seule validation à la fin du traitement des données pour la totalité de l'ensemble. L'avantage de l'utilisation de ce paramètre COMMIT_COUNT est que le traitement peut s'achever plus rapidement.</p> <p><i>Entier non NULL</i></p> <p>Le programme Apply traite l'ensemble d'abonnements en mode transactionnel. Après extraction de tous les ensembles de réponses, les contenus des fichiers auxiliaires seront appliqués dans l'ordre de la séquence de validation, classant chaque transaction par valeur IBMSNAP_INTENTSEQ. Ce type de traitement permet d'ouvrir et de traiter simultanément tous les fichiers auxiliaires. Une validation sera émise en fonction du nombre de transactions spécifié dans cette colonne. Par exemple, 1 signifie une validation après chaque transaction, 2 signifie une validation après deux transactions, etc. Un entier de 0 signifie qu'une seule validation sera émise après l'application de toutes les données extraites. L'avantage de l'utilisation du traitement en mode transactionnel est que le traitement tient compte des contraintes d'intégrité référentielle sur la cible et que des validations provisoires peuvent être émises.</p> <p>Le mode transaction change le comportement du programme Apply uniquement pour les ensembles possédant des tables cible de type CCD, copie utilisateur et point de cohérence. Les ensembles contenant des tables réplique sont toujours traités en mode transaction.</p>
MAX_SYNCH_MINUTES	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : oui</p> <p>Limite de seuil de temps régulant la quantité de données de modification à extraire et à appliquer lors d'un cycle d'abonnement. Le programme Apply divise le traitement de l'ensemble d'abonnements en mini-cycles en fonction de la colonne IBMSNAP_LOGMARKER de la table UOW ou de la table de modification cohérente des données sur le serveur de Capture et émet une validation sur le serveur cible après chaque mini-cycle réussi. La limite est automatiquement recalculée si le programme Apply rencontre une contrainte de ressource rendant la limite définie impossible. Les valeurs MAX_SYNCH_MINUTES inférieures à 1 seront traitées de la même manière qu'une MAX_SYNCH_MINUTES égale à Null.</p>
AUX_STMTS	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non</p> <p>Nombre d'instructions SQL que vous définissez dans la table IBMSNAP_SUBS_STMTS pouvant être exécutées avant ou après le traitement d'un ensemble d'abonnements par le programme Apply.</p>

Tableau 95. Colonnes dans la table IBMSNAP\_SUBS\_SET (suite)

Nom de colonne	Description
ARCH_LEVEL	<p><b>Type de données</b> : CHAR(4) ; <b>Null admis</b> : non</p> <p>Niveau d'architecture des tables de contrôle de réplication. Cette colonne identifie les règles sous lesquelles une ligne a été créée. Ce niveau est défini par IBM.</p> <p><b>0801</b> Version 8 ou ultérieure de la réplication SQL</p> <p><b>0803</b> Version 8 de la réplication SQL avec prise en charge des sources Oracle</p> <p><b>0805</b> Version 8 de la réplication SQL avec prise en charge de DB2 pour le mode nouvelle fonction z/OS</p>

## Table ASN.IBMSNAP\_SUBS\_STMTS

La table IBMSNAP\_SUBS\_STMTS contient les instructions SQL ou les appels de procédures stockées définis par l'utilisateur qui seront exécutés avant ou après chaque cycle de traitement d'ensemble d'abonnements. Les instructions Déclarer immédiatement (EI) ou les procédures stockées peuvent être exécutées sur le serveur source ou cible uniquement. Cette table est remplie lorsque vous définissez un ensemble d'abonnements utilisant des instructions SQL ou des appels de procédures stockées.

**Serveur** : serveur de contrôle Apply

**Index** : APPLY\_QUAL, SET\_NAME, WHOS\_ON\_FIRST, BEFORE\_OR\_AFTER, STMT\_NUMBER

**Important** : Faites attention lorsque vous mettez à jour cette table avec SQL. La modification inopportune de cette table peut entraîner des résultats imprévus et des pertes de données. Le nombre d'entrées pour un abonnement doit être reflété dans la colonne AUX\_STMTS de la table IBMSNAP\_SUBS\_SET. Si AUX\_STMTS est égal à zéro pour un ensemble d'abonnements, les entrées correspondantes dans la table IBMSNAP\_SUBS\_STMTS sont ignorées par le programme Apply.

Le tableau 96 fournit une brève description des colonnes dans la table IBMSNAP\_SUBS\_STMTS.

Tableau 96. Colonnes dans la table IBMSNAP\_SUBS\_STMTS

Nom de colonne	Description
APPLY_QUAL	<p><b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non</p> <p>Identifie uniquement quel programme Apply traite l'instruction SQL ou la procédure stockée.</p>
SET_NAME	<p><b>Type de données</b> : CHAR(18); <b>Valeur NULL admise</b> : non</p> <p>Nom de l'ensemble d'abonnements auquel cette instruction SQL ou cette procédure stockée est associée.</p>

Tableau 96. Colonnes dans la table IBMSNAP\_SUBS\_STMTS (suite)

Nom de colonne	Description
WHOS_ON_FIRST	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Les valeurs suivantes sont utilisées pour contrôler l'ordre de traitement dans les scénarios de réplication bidirectionnelle.</p> <p><b>F</b> (premier) La table cible est la table utilisateur ou la réplique parente. La table source est la réplique dépendante et, dans le cas d'un conflit de mises à jour entre la table source et la table cible, les transactions conflictuelles de la table source seront rejetées. F n'est pas utilisé pour les abonnements en lecture seule.</p> <p><b>S</b> (deuxième) La table source est la table utilisateur, la réplique parente ou une autre source. La table cible est la réplique dépendante ou une autre copie et, dans le cas d'un conflit de mises à jour entre la table source et la table cible, les transactions conflictuelles de la table cible seront rejetées. S est utilisé pour tous les abonnements en lecture seule.</p>
BEFORE_OR_AFTER	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Valeur indiquant quand et où l'instruction est émise :</p> <p><b>A</b> L'instruction est exécutée sur le serveur cible après l'application de toutes les lignes définies par la réponse.</p> <p><b>B</b> L'instruction est exécutée sur le serveur cible avant l'application des lignes définies par la réponse.</p> <p><b>S</b> L'instruction est exécutée sur le serveur de contrôle de Capture avant l'ouverture des curseurs définis par la réponse.</p> <p><b>G</b> Réservé à une utilisation par la réplication SQL.</p> <p><b>X</b> Réservé à une utilisation par la réplication SQL.</p>
STMT_NUMBER	<p><b>Type de données</b> : SMALLINT ; <b>Null admis</b> : non</p> <p>Définit l'ordre relatif d'exécution dans les limites de la valeur de la colonne BEFORE_OR_AFTER.</p>
EI_OR_CALL	<p><b>Type de données</b> : CHAR(1) ; <b>Null admis</b> : non</p> <p>Valeur indiquant :</p> <p><b>E</b> L'instruction SQL doit être exécutée comme EXEC SQL EXECUTE IMMEDIATE.</p> <p><b>C</b> L'instruction SQL contient un nom de procédure stockée à exécuter comme EXEC SQL CALL.</p>
SQL_STMT	<p><b>Type de données</b> : VARCHAR(1024) ; <b>Null admis</b> : oui</p> <p>Une des valeurs suivantes :</p> <p><b>Instruction</b> L'instruction SQL doit être exécutée comme une instruction EXEC SQL EXECUTE IMMEDIATE si EI_OR_CALL est E.</p> <p><b>Procédure.</b> Nom sur huit octets d'une procédure stockée SQL, sans paramètre, ou le mot clé CALL exécuté comme une instruction EXEC SQL CALL si EI_OR_CALL est C.</p>



Tableau 96. Colonnes dans la table IBMSNAP\_SUBS\_STMTS (suite)

Nom de colonne	Description
ACCEPT_SQLSTATES	<p><b>Type de données</b> : VARCHAR(50) ; <b>Null admis</b> : oui</p> <p>Une à dix valeurs sur cinq octets SQLSTATE que vous avez spécifiées lorsque vous avez défini l'ensemble d'abonnements. Ces valeurs différentes de zéro sont acceptées par le programme Apply comme une exécution réussie. Toute autre valeur entraînera l'échec de l'exécution.</p>

## Tables sur le serveur de contrôle cible

Différents types de tables cible sont stockées sur le serveur cible. Si vous n'utilisez pas une table existante comme table cible, le programme de ligne de commande ASNCLP ou le centre de réplication crée la table cible d'après vos spécifications, en fonction de la manière dont vous définissez le membre d'ensemble d'abonnements.

Le tableau 97 décrit les tables sur le serveur cible.

Tableau 97. Guide de référence pour les tables cible

Nom de la table	Description
«Table d'agrégation de base»	Contient des données agrégées issues d'une table source.
«Table d'agrégation des modifications», à la page 486	Contient des données agrégées issues d'une table CD.
«Cibles CCD», à la page 87	Contient des informations sur les changements qui se produisent dans la source et contient des colonnes supplémentaires pour identifier l'ordre séquentiel de ces changements.
«Table des points de cohérence», à la page 489	Une copie des données source, avec une colonne supplémentaire qui enregistre l'heure à laquelle les données ont été validées dans le journal source.
«Table réplique», à la page 489	Un type de table cible utilisé pour la réplication d'égal à égal (update-anywhere).
«Table de copie utilisateur», à la page 490	Une copie de la table source.

### Table d'agrégation de base

Une table d'agrégation de base est une table cible qui contient les résultats des fonctions d'agrégation exécutées sur les données situées dans la table source.

*schéma.agrégat\_base*

**Serveur** : serveur cible

**Important** : Si vous utilisez une instruction SQL pour mettre à jour cette table, vous risquez de perdre vos mises à jour si une actualisation complète est effectuée par le programme Apply.

Le tableau 98, à la page 486 fournit une brève description des colonnes dans la table d'agrégation de base.

Tableau 98. Colonnes dans la table d'agrégation de base

Nom de colonne	Description
<i>colonnes utilisateur</i>	Les données globales qui ont été calculées à partir de la table source.
IBMSNAP_LLOGMARKER	L'horodatage actuel sur le serveur source au moment où l'agrégation des données dans la table source a commencé.
IBMSNAP_HLOGMARKER	L'horodatage actuel sur le serveur source au moment où l'agrégation des données dans la table source s'est terminée.

## Table d'agrégation des modifications

Une table d'agrégation des modifications est une table cible qui contient les résultats des fonctions d'agrégation exécutées sur les données dans la table CD (change-data). Cette table est similaire à la table d'agrégation de base, à cette différence près que les fonctions exécutées sur la table CD concernent uniquement des modifications qui se produisent dans un intervalle de temps spécifique.

*schéma.agrégat\_modification*

**Serveur** : serveur cible

**Important** : Si vous utilisez une instruction SQL pour mettre à jour cette table, vous risquez de perdre vos mises à jour si une actualisation complète est effectuée par le programme Apply.

Le tableau 99 fournit une brève description des colonnes dans la table d'agrégation des modifications.

Tableau 99. Colonnes dans la table d'agrégation des modifications

Nom de colonne	Description
<i>colonnes clé utilisateur</i>	Les colonnes qui constituent la clé cible.
<i>colonnes non clé utilisateur</i>	Les colonnes de données non clé issues de la table source. Les noms de colonne dans cette table cible ne doivent pas nécessairement correspondre aux noms de colonne dans la table source, mais les types de données doivent correspondre.
<i>colonnes définies par l'utilisateur</i>	Colonnes définies par l'utilisateur, dérivées d'expressions SQL. Vous pouvez utiliser ce type de colonne avec des fonctions SQL pour convertir des types de données source en différents types de données cible.
IBMSNAP_LLOGMARKER	La valeur IBMSNAP_LOGMARKER ou IBMSNAP_LLOGMARKER la plus ancienne dans les lignes des tables (CD+UOW) ou CCD qui font l'objet de l'agrégation.
IBMSNAP_HLOGMARKER	La valeur IBMSNAP_LOGMARKER ou IBMSNAP_HLOGMARKER la plus récente dans les lignes des tables (CD+UOW) ou CCD qui font l'objet de l'agrégation.

## Cibles CCD

Les tables cible CCD (Consistent-Change-Data) fournissent les données de transactions validées qui peuvent être lues et utilisées par d'autres applications, par exemple InfoSphere DataStage. Vous pouvez également utiliser une table de modification cohérente des données (CCD) pour effectuer un audit des données source ou conserver un historique sur l'utilisation des données.

Par exemple, vous pouvez suivre les comparaisons avant et après des données, le moment où des modifications sont apportées ainsi que l'ID utilisateur ayant apporté la modification dans la table source.

Pour définir une table cible en lecture seule qui conserve un historique de la table source, définissez la table CCD cible de telle sorte qu'elle possède les attributs suivants :

#### **Non condensée**

Pour conserver un enregistrement de toutes les modifications source, définissez la table CCD de telle sorte qu'elle soit non condensée, c'est-à-dire qu'elle possède une ligne par modification apportée. Les tables non condensées contiennent de nombreuses lignes portant la même valeur clé ; par conséquent, ne définissez pas un index unique. Une table CCD non condensée contient une ligne par opération UPDATE, INSERT ou DELETE, ce qui permet de constituer un historique des opérations effectuées dans la table source. Si vous capturez des opérations UPDATE en tant qu'opérations INSERT et DELETE (pour le partitionnement de colonnes), la table CCD contiendra deux lignes par mise à jour (une ligne pour l'opération DELETE et une ligne pour l'opération INSERT).

#### **Complète ou incomplète**

Vous pouvez déterminer si vous souhaitez que la table CCD soit complète ou incomplète. Les tables CCD incomplètes ne contiennent pas à l'origine d'ensemble complet de lignes source. Vous devez donc créer une table CCD incomplète pour pouvoir conserver un historique des mises à jour apportées à une table source (mises à jour effectuées depuis que le programme Apply a commencé à remplir la table CCD).

#### **Inclure des colonnes UOW**

Pour les fonctions d'audit avancées, incluez les colonnes supplémentaires de la table UOW. Si vous avez besoin d'une identification davantage orientée utilisateur, des colonnes pour ID de corrélation et ID d'autorisation primaire pour DB2 pour z/OS, ou encore pour nom de travail et profil utilisateur System i sont disponibles dans la table UOW.

**Important pour la version 10.1 sous Linux, UNIX et Windows :** Si la base de données source est DB2 10.1 sous Linux, UNIX et Windows avec des noeuds multiples DB2 pureScale, les cibles CCD sont prises en charge seulement si les programmes Capture et Apply ont été migrés vers la version 10.1 et si le paramètre de **compatibility** Capture est défini sur 1001. Avec un seul noeud DB2 pureScale et **compatibility** défini sur 0801, les cibles CCD sont prises en charge même si la version du programme Apply est antérieure à 10.1.

Par définition, une table CCD comprend toujours les colonnes suivantes en plus des colonnes répliquées à partir de la table source :

Colonne	Description
IBMSNAP_INTENTSEQ	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : non</p> <p>Un numéro de séquence qui identifie uniquement un changement. Cette valeur est croissante dans une transaction.</p> <p><b>z/OS</b> Numéro de séquence du journal (LRSN ou RBA) de chaque opération de mise à jour, de suppression et d'insertion.</p>
IBMSNAP_OPERATION	<p><b>Type de données</b> : CHAR(1); <b>Null admis</b> : Non</p> <p>Indicateur qui spécifie le type d'opération : I (INSERT), U (UPDATE) ou D (DELETE).</p>
IBMSNAP_COMMITSEQ	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : non</p> <p>Numéro de séquence pour chaque ligne au sein d'une transaction.</p> <p><b>z/OS</b> Numéro de séquence du journal (LRSN ou RBA) de l'enregistrement de validation source.</p>
IBMSNAP_LOGMARKER	<p><b>Type de données</b> : TIMESTAMP; <b>Null admis</b> : Non</p> <p>L'heure approximative à laquelle les données ont été validées.</p>

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><b>Type de données</b> : VARCHAR(30); <b>Null admis</b> : Oui</p> <p>ID utilisateur qui a mis à jour la table source.</p> <p><b>z/OS</b> Cette colonne correspond à l'ID utilisateur primaire.</p>
IBMSNAP_AUTHTKN	<p><b>Type de données</b> : VARCHAR(30); <b>Null admis</b> : Oui</p> <p>Clé d'accès qui est associée à la transaction.</p> <p><b>z/OS</b> ID de corrélation (normalement un nom de travail) qui a exécuté la mise à jour source.</p>

Colonne	Description
IBMSNAP_PLANID	<p><b>Type de données</b> : VARCHAR(8); <b>Null admis</b> : Oui</p> <p><b>z/OS</b> Nom de plan associé à la transaction. Cette colonne doit être NULL pour DB2 for Linux, UNIX et Windows.</p>
IBMSNAP_UOWID	<p><b>Type de données</b> : VARCHAR(16) FOR BIT DATA ; <b>Null admis</b> : oui</p> <p>Identificateur de l'unité de travail (UOW) provenant de l'enregistrement de journal pour une ligne.</p> <p><b>z/OS</b> Identificateur de l'unité de travail, parfois appelé ID unité de récupération (URID) de la transaction.</p>

## Table des points de cohérence

La table des points de cohérence contient une copie des données source, ainsi qu'une colonne système supplémentaire (IBMSNAP\_LOGMARKER) contenant un horodatage qui indique approximativement le moment où la ligne a été insérée ou mise à jour sur le serveur source.

*schéma.points\_de\_cohérence*

**Serveur** : serveur cible

**Important** : Si vous utilisez une instruction SQL pour mettre à jour cette table, vous risquez de perdre vos mises à jour si une actualisation complète est effectuée par le programme Apply.

Le tableau 100 fournit une brève description des colonnes dans la table des points de cohérence.

Tableau 100. Colonnes dans la table des points de cohérence

Nom de colonne	Description
<i>colonnes clé utilisateur</i>	Les colonnes qui constituent la clé cible.
<i>colonnes non clé utilisateur</i>	Les colonnes de données non clé issues de la vue ou de la table source. Les noms de colonne dans cette table cible ne doivent pas nécessairement correspondre aux noms de colonne dans la table source, mais les types de données doivent correspondre.
<i>colonnes définies par l'utilisateur</i>	Colonnes définies par l'utilisateur, dérivées d'expressions SQL. Vous pouvez utiliser ce type de colonne avec des fonctions SQL pour convertir des types de données source en différents types de données cible.
IBMSNAP_LOGMARKER	L'heure approximative de validation sur le serveur de contrôle Capture. La valeur de cette colonne est null après une actualisation complète.

## Table réplique

La table réplique doit avoir les mêmes colonnes clé que la table source. En raison de ces similitudes, la table réplique peut être utilisée comme table source pour d'autres ensembles d'abonnements. La conversion d'une table cible en table source

est effectuée automatiquement lorsque vous définissez un type de cible réplique et que vous spécifiez l'attribut CHANGE DATA CAPTURE.

*schéma.réplique*

**Serveur** : serveur cible

Cette table contient des informations que vous pouvez mettre à jour à l'aide de SQL.

Le tableau 101 fournit une brève description des colonnes dans la table réplique.

Tableau 101. Colonnes dans la table réplique

Nom de colonne	Description
<i>colonnes clé utilisateur</i>	Les colonnes qui constituent la clé cible, qui doit être la clé primaire de la table maître.
<i>colonnes non clé utilisateur</i>	Les colonnes de données non clé issues de la table source. Les noms de colonne dans cette table cible ne doivent pas nécessairement correspondre aux noms de colonne dans la table source, mais les types de données doivent correspondre.

## Table de copie utilisateur

La table de copie utilisateur est une table cible qui contient une copie des colonnes figurant dans la table source. Cette table cible peut être un sous-ensemble de lignes ou de colonnes issu de la table source, mais elle ne peut pas contenir des colonnes supplémentaires.

*schéma.copie\_utilisateur*

**Serveur** : serveur cible

**Important** : Si vous utilisez une instruction SQL pour mettre à jour cette table, vous risquez de perdre vos mises à jour si une actualisation complète est effectuée par le programme Apply.

Exception faite de la création de sous-ensembles et de l'amélioration des données, une table de copie utilisateur reflète un état valide de la table source, mais pas nécessairement l'état actuel. Les références à des tables de copie utilisateur (ou à d'autres types de tables cible) limitent les risques de conflits potentiels en cas de nombre élevé d'accès direct aux tables source. L'accès à des tables de copie utilisateur locales est un processus beaucoup plus rapide que l'accès à des tables source distantes via le réseau pour chaque requête.

Le tableau 102 fournit une brève description des colonnes de la table de copie utilisateur.

Tableau 102. Colonnes de la table de copie utilisateur

Nom de colonne	Description
<i>colonnes clé utilisateur</i>	Les colonnes qui constituent la clé cible.
<i>colonnes non clé utilisateur</i>	Les colonnes de données non clé issues de la vue ou de la table source. Les noms de colonne dans cette table cible ne doivent pas nécessairement correspondre aux noms de colonne dans la table source, mais les types de données doivent correspondre.

Tableau 102. Colonnes de la table de copie utilisateur (suite)

Nom de colonne	Description
<i>colonnes définies par l'utilisateur</i>	Colonnes définies par l'utilisateur, dérivées d'expressions SQL. Vous pouvez utiliser ce type de colonne avec des fonctions SQL pour convertir des types de données source en différents types de données cible.





---

## Annexe A. Schémas de codage UNICODE et ASCII pour la réplication SQL (z/OS)

La réplication SQL pour OS/390 et z/OS Version 7 ou ultérieure prend en charge les schémas de codage UNICODE et ASCII.

Pour exploiter le schéma de codage UNICODE, vous devez avoir au moins DB2 for OS/390 and z/OS version 7 et vous devez créer ou convertir manuellement vos tables source, cible et de contrôle de réplication SQL comme décrit dans les sections suivantes. Cependant, votre environnement de réplication existant fonctionnera avec la réplication SQL pour OS/390 et z/OS Version 7 ou ultérieure, même si vous ne modifiez pas les schémas de codage. Si votre système est un système UNICODE, vous devez ajouter ENCODING(EBCDIC) aux commandes BIND PLAN et PACKAGE pour les programmes Capture, Apply, et moniteur d'alertes de réplication.

---

### Règles pour la sélection d'un schéma de codage

Si vos tables source, de données de modification, et cible utilisent le même schéma de codage, vous pouvez réduire la nécessité de conversions de données dans votre environnement de réplication.

Lorsque vous sélectionnez des schémas de codage pour les tables, suivez la règle CCSID unique :

Les données d'espace table sont codées à l'aide des CCSID ASCII, EBCDIC ou UNICODE. Le schéma de codage de toutes les tables référencées par une instruction SQL doit être identique. De même, toutes les tables que vous utilisez dans les vues et les jointures doivent utiliser le même schéma de codage.

Si vous ne suivez pas la règle de CCSID unique, DB2 détectera la violation et renverra un message SQLCODE -873 pendant la liaison ou l'exécution.

Le choix de ASCII ou UNICODE pour vos tables dépend de la configuration de votre client/serveur. Plus spécifiquement, suivez ces règles lorsque vous choisissez des schémas de codage pour les tables :

- Les tables source ou cible sur DB2 for OS/390 peuvent être EBCDIC, ASCII, ou UNICODE. Elles peuvent être copiées à partir de ou dans des tables dont le schéma de codage est identique ou différent, dans tout DBMS pris en charge (famille DB2 ou non DB2 avec DataJoiner).
- Sur un serveur source DB2 for OS/390, les tables de données de modifications et d'unités d'oeuvre sur le même serveur n'ont pas à utiliser le même schéma de codage si, lorsque le membre de l'ensemble d'abonnements est créé, le type cible est USERCOPY et JOIN\_UOW\_CD n'est pas égal à Y. Sinon, les tables de données de modification et d'unités d'oeuvre doivent utiliser le même schéma de codage.
- La table IBMSNAP\_SIGNAL doit être codée EBCDIC afin que le programme Capture n'ait pas à traduire des signaux en EBCDIC lorsqu'il les sélectionne dans la table du signal.
- Toutes les tables de contrôle (ASN.IBMSNAP\_SUBS\_XXXX) sur le même serveur de contrôle doivent utiliser le même schéma de codage.

- Les autres tables de contrôle peuvent utiliser tout schéma de codage.

---

## Paramétrage des schémas de codage

Pour spécifier le schéma de codage approprié pour les tables, modifiez le SQL qui est utilisé pour générer les tables.

Il est recommandé d'arrêter les programmes Capture et Apply avant de modifier le schéma de codage des tables existantes.

**Remarque :** *DB2 for z/OS V8 SQL Reference* contient de plus amples informations sur CCSID.

Pour paramétrer les schémas de codage :

1. Créez de nouvelles tables source et cible avec le schéma de codage adéquat. Il est ensuite recommandé d'initialiser Capture par un démarrage à froid et de redémarrer le programme Apply.
2. Si vous avez déjà créé vos tables source et cible, modifiez les schémas de codage des tables source et cible existantes. Les tables existantes doivent avoir le même schéma de codage dans un espace table.
  - a. Utilisez la fonctionnalité Réorg. l'espace table pour décharger l'espace table existant.
  - b. Supprimez l'espace table existant.
  - c. Re-créez l'espace table en spécifiant le nouveau schéma de codage.
  - d. Utilisez la fonctionnalité Charger pour charger les anciennes données dans le nouvel espace table. Voir *DB2 for z/OS V8 Utility Guide and Reference* pour obtenir plus d'informations sur les fonctionnalités Charger et Réorg.
3. Utilisez le centre de réplication pour créer de nouvelles tables de contrôle avec le schéma de codage adéquat.
4. Utilisez les fonctionnalités Réorg. et Charger pour modifier le schéma de codage des tables de contrôle existantes et des tables de données de modification.
5. Lorsque vous créez de nouvelles sources de réplication ou de nouveaux ensembles d'abonnements à l'aide de ASNCLP ou du Centre de réplication, indiquez le schéma de codage adéquat.

---

## Annexe B. Démarrage des programmes de réplication SQL depuis une application (Linux, UNIX, Windows)

Vous pouvez démarrer un programme de réplication (programme Capture, programme Apply, moniteur d'alertes de réplication) pour un cycle de réplication en appelant des routines depuis votre application.

Pour pouvoir utiliser ces routines, vous devez spécifier l'option AUTOSTOP pour le programme Capture et l'option COPYONCE pour le programme Apply, car l'interface de programme d'application (API) ne prend en charge que l'exécution synchrone.

Des exemples d'API et leurs fichiers makefile respectifs se trouvent dans les répertoires suivants :

### Windows

sqllib\samples\repl

### Linux UNIX

sqllib/samples/repl

Ces répertoires contiennent les fichiers suivants, utilisés pour le démarrage du programme Capture :

#### **capture\_api.c**

Code exemple utilisé pour le démarrage du programme Capture sous Windows, Linux ou UNIX.

#### **capture\_api\_nt.mak**

Fichier makefile du code exemple sous Windows.

#### **capture\_api\_unix.mak**

Fichier makefile du code exemple sous UNIX.

Ces répertoires contiennent les fichiers suivants, utilisés pour le démarrage du programme Apply :

#### **apply\_api.c**

Code exemple utilisé pour le démarrage du programme Apply sous Windows, Linux ou UNIX.

#### **apply\_api\_nt.mak**

Fichier makefile du code exemple sous Windows.

#### **apply\_api\_unix.mak**

Fichier makefile du code exemple sous UNIX.

Ces répertoires contiennent les fichiers suivants, utilisés pour le démarrage du moniteur d'alertes de réplication :

#### **monitor\_api.c**

Code exemple utilisé pour le démarrage du moniteur d'alertes de réplication sous Windows, Linux ou UNIX.

#### **monitor\_api\_nt.mak**

Fichier makefile du code exemple sous Windows.

**monitor\_api\_unix.mak**

Fichier makefile du code exemple sous UNIX.

## Annexe C. Comment le programme Capture traite les types d'entrée de journal pour la réplication SQL (System i)

La table suivante décrit comment le programme Capture traite différents types d'entrée de journal.

Tableau 103. Traitement du programme Capture par entrée de journal

Code journal <sup>1</sup>	Type d'entrée	Description	Traitement
C	CM	Ensemble de modifications d'enregistrement validées	Insérez un enregistrement dans la table des unités d'oeuvre (UOW).
C	RB	Annulation	Aucune ligne UOW insérée.
F	AY	Modifications journalisées appliquées au membre de fichier physique	Emettez un message ASN2004 et la régénération intégrale du fichier.
F	CE	Modifiez la fin des données pour le fichier physique	Emettez un message ASN2004 et la régénération intégrale du fichier.
F	CR	Membre de fichier physique effacé	Emettez un message ASN2004 et la régénération intégrale du fichier.
F	EJ	Fin de la journalisation du membre de fichier physique	Emettez un message ASN200A et la régénération intégrale du fichier. Une régénération intégrale se produit chaque fois que le programme Capture lit une entrée de journal EJ, que l'utilisateur ou le système ait demandé la fin de la journalisation. Consultez la documentation appropriée sur System i pour obtenir des informations sur les événements implicites de fin de journal pour un fichier.
F	IZ	Membre de fichier physique initialisé	Emettez un message ASN2004 et la régénération intégrale du fichier.
F	MD	Membre supprimé du fichier physique (DLTLIB, DLTF, ou RMVM)	Emettez un message ASN200A et tentez une régénération intégrale.
F	MF	Stockage pour un membre de fichier physique libéré	Emettez un message ASN2004 et la régénération intégrale du fichier.
F	MM	Fichier physique contenant un membre déplacé (renommer l'objet(RNMOBJ) de la bibliothèque, Déplacer l'objet (MOVOBJ) du fichier)	Emettez un message ASN200A et tentez une régénération intégrale.

Tableau 103. Traitement du programme Capture par entrée de journal (suite)

Code journal <sup>1</sup>	Type d'entrée	Description	Traitement
F	MN	Fichier physique contenant un membre renommé(RNMOBJ de fichier, Renommer membre (RNMM))	Emettez un message ASN200A et tentez une régénération intégrale.
F	MR	Membre de fichier physique restauré	Emettez un message ASN2004 et la régénération intégrale du fichier.
F	RC	Modifications journalisées supprimées du membre de fichier physique	Emettez un message ASN2004 et la régénération intégrale du fichier.
F	RG	Membre de fichier physique réorganisé	Si le numéro relatif d'enregistrement de la table source est utilisé comme clé de réplication, émettez un message ASN2004 et une régénération intégrale du fichier.
J	NR	Identification des prochains récepteurs de journal	Réinitialisez le programme Capture.
J	PR	Identificateur des précédents récepteurs de journal	Incrémentez le compteur de numéro de séquence unique.
R	DL	Enregistrement supprimé du membre de fichier physique	Insérez un enregistrement DLT dans la table de données de modification (CD).
R	DR	Enregistrement supprimé pour l'annulation	Insérez un enregistrement DLT dans la table de données de modification (CD).
R	PT	Enregistrement ajouté au membre de fichier physique	Insérez un enregistrement ADD dans la table de données de modification (CD).
R	PX	Enregistrement ajouté directement au membre de fichier physique	Insérez un enregistrement ADD dans la table de données de modification (CD).
R	UB	Image avant de l'enregistrement mise à jour dans le membre de fichier physique	Voir la note 2.
R	UP	Image après de l'enregistrement mise à jour dans le membre de fichier physique	Voir la note 2.
R	BR	Image avant de l'enregistrement mise à jour pour l'annulation	Voir la note 3.
R	UR	Image après de l'enregistrement mise à jour pour l'annulation	Voir la note 3.

Tableau 103. Traitement du programme Capture par entrée de journal (suite)

Code journal <sup>1</sup>	Type d'entrée	Description	Traitement
<b>Remarque :</b>			
1. Les valeurs suivantes sont utilisées pour les codes journal :			
	<b>C</b>	Opération de contrôle de validation	
	<b>F</b>	Opération de fichier de base de données	
	<b>J</b>	Opération de journal ou de récepteur de journal	
	<b>R</b>	Opération sur un enregistrement spécifique	
2. L'image R-UP et l'image R-UB forment un seul enregistrement UPD dans la table de données de modification si la colonne PARTITION_KEYS_CHG dans la table de registres est N. Sinon, l'image R-UB insère un enregistrement DLT dans la table de données de modification et l'image R-UP insère un enregistrement ADD dans la table de données de modification.			
3. L'image R-UR et l'image R-BR forment un seul enregistrement UPD dans la table de données de modification si la colonne PARTITION_KEYS_CHG dans la table de registres est N. Sinon, l'image R-BR insère un enregistrement DLT dans la table de données de modification et l'image R-UR insère un enregistrement ADD dans la table de données de modification.			

Tous les autres types d'entrée de journal sont ignorés par le programme Capture.





---

## 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 104. 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 <a href="http://www.ibm.com/support/entry/portal/Software/Information_Management/InfoSphere_Information_Server">www.ibm.com/support/entry/portal/Software/Information_Management/InfoSphere_Information_Server</a>
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 <a href="http://www.ibm.com/businesssolutions/">www.ibm.com/businesssolutions/</a>
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 <a href="http://www.ibm.com/account/">www.ibm.com/account/</a>
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 <a href="http://www.ibm.com/software/sw-training/">http://www.ibm.com/software/sw-training/</a>
Interlocuteurs IBM	Pour en savoir plus sur les solutions, vous pouvez contacter un interlocuteur IBM à l'adresse <a href="http://www.ibm.com/connect/ibm/us/en/">www.ibm.com/connect/ibm/us/en/</a>

## 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/](http://www.ibm.com/software/data/integration/support/federation_server/)
- IBM InfoSphere Replication Server  
[www.ibm.com/software/data/integration/support/replication\\_server/](http://www.ibm.com/software/data/integration/support/replication_server/)
- IBM InfoSphere Data Event Publisher  
[www.ibm.com/software/data/integration/support/data\\_event\\_publisher/](http://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/](http://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/](http://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/](http://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/](http://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 105. 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 <a href="http://www.ibm.com/software/data/info/consumability-survey">www.ibm.com/software/data/info/consumability-survey</a>
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"><li>• Formulaire de commentaire en ligne : <a href="http://www.ibm.com/software/data/rcf/">www.ibm.com/software/data/rcf/</a></li><li>• Courriel : <a href="mailto:comments@us.ibm.com">comments@us.ibm.com</a></li></ul>

---

## Lecture des diagrammes de syntaxe

Les règles suivantes s'appliquent aux diagrammes de syntaxe utilisés dans ce document :

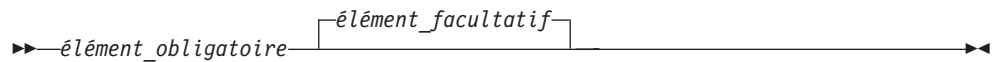
- Lisez les diagrammes de syntaxe de gauche à droite, de haut en bas en suivant le chemin de la ligne. Les conventions suivantes sont utilisées :
  - Le symbole >>--- indique le début d'un diagramme de syntaxe.
  - Le symbole ---> indique que le diagramme de syntaxe continue sur la ligne suivante.
  - 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.

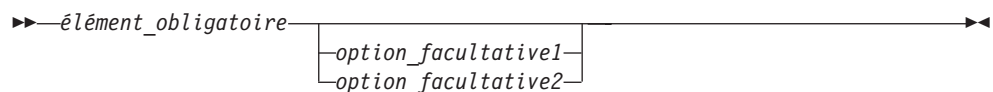


- Si vous pouvez faire une sélection parmi plusieurs éléments, ceux-ci apparaissent à la verticale, dans une pile.

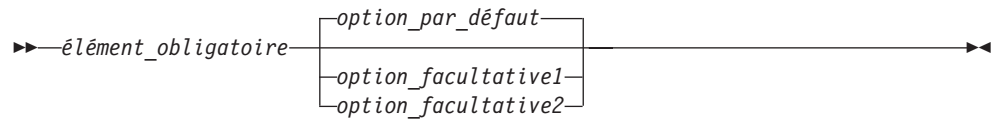
Si vous devez sélectionner l'un des éléments, un seul élément de la pile apparaît dans le chemin principal.



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](http://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](http://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 Etats-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 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.

CASS, CASS Certified, DPV, LACSLink, 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 LACSLink 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](http://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

- à intervalles réguliers 78
- abonnement à des sources 71
- activation
  - ensembles d'abonnements 74
- actualisation
  - tables uniques 166
- administration
  - autorisations nécessaires 15
- ajustement
  - performances 12
- Analyseur
  - pour System i
    - paramètres d'appel 352
  - System i
    - création de modules SQL 33
    - paramètres d'appel 352
- applications
  - démarrage de programmes de réplification à partir de 495
- ARM (Automatic Restart Manager) 174
- arrêt
  - programme Apply
    - System i 156, 360
    - UNIX 156
    - Windows 156
    - z/OS 156
  - Programme Capture
    - System i 137, 363
    - UNIX 137
    - Windows 137
    - z/OS 137
- arrêt de la capture des modifications 181
- asnpwd 281
- anscr 285
- asndrop 288
- asntc 305
- attribut DATA CAPTURE CHANGES 23
- attributs
  - modification pour des ensembles d'abonnements 188
  - modification pour des objets enregistrés 178
- autorisations
  - pour l'administration 15
  - pour le programme Apply 18
  - pour le programme Capture 17
  - pour les déclencheurs Capture 19

## C

- Capture
  - plusieurs partitions de base de données 34
  - utilisation de plusieurs partitions de base de données 29
- Capture, programme
  - commandes
    - asncap 257

- Capture, programme (*suite*)
  - pour System i
    - paramètre RESTART 394
    - paramètres de démarrage à chaud 394
    - paramètres de démarrage à froid 394
  - pour UNIX
    - paramètre add\_partition 123
  - pour Windows
    - paramètre add\_partition 123
  - pour z/OS
    - paramètre add\_partition 123
- caractères de fin
  - scripts SQL générés 253
- Centre de réplification
  - communication avec
    - déclencheurs de capture 243
    - moniteur d'alertes de réplification 247
  - programme Apply 243
  - Programme Capture 243
  - connectivité 24
  - fonctions de promotion 208
- changement de nom
  - colonnes 98, 115
- clause WHERE
  - restriction de la colonne PREDICATES 111
  - sous-ensembles de lignes 97
- clés cible
  - présentation 99
- clés de partitionnement logique
  - description 55
- clés primaires
  - numéros relatifs d'enregistrement
    - System i 63
  - partitionnement logique 55
  - utilisé comme clé cible 99
- clients, service
  - contact 501
- cohérence des données 94
- cold startmode 122
- colonne JOIN\_UOW\_CD 111
- colonne MAX\_SYNCH\_MINUTES
  - mise en bloc de données 74
- colonne PREDICATES 111
- colonne UOW\_CD\_PREDICATES 111
- colonnes
  - ajout à des tables source enregistrées 178
  - calculé 96
  - calculées 115
  - changement de nom 98, 115
  - définition dans la table cible 96
  - définition de sous-ensembles
    - dans la cible 96
    - sur la source 49
  - disponibles pour la réplification 49
  - enregistrement dans la table source 49

- colonnes (*suite*)
  - image après 51
  - image avant 51
  - mappage entre sources et cibles 98
  - numéros relatifs d'enregistrement sur System i 63
- colonnes calculées 96
  - création 115
  - table CD 86
  - table source 86
- colonnes d'identificateurs de ligne
  - type de données LOB 104
- colonnes d'identité 107
- colonnes d'image après
  - colonnes d'image-avant dans la réplification Q 51
- colonnes de clé cible
  - mise à jour 100
- colonnes de clés primaires mises à jour 55
- colonnes GENERATED ALWAYS 107
- colonnes image-avant
  - description 51
  - restrictions 51
  - tables d'agrégation des modifications 97
- commande \$TA JES2 238
- commande ADDDPREG
  - description 317
- commande ADDDPRSUB 326
- commande ADDDPRSUBM 341
- commande ADDJOBSCDE 238
- commande ANZDPR 352
- commande ANZDPRJRN 44
- commande asnacd 277
- commande asnanalyze 278
- commande asnaply 270
- commande asncap
  - démarrage de capture 257
- commande asnccmd
  - fonctionnement de capture 266
- commande asnpwd
  - gestion de l'accès aux serveurs distants 20
- commande asnlist 289
- commande asntdiff 229, 290, 294, 302
  - exécution en mode parallèle 229
- commande asntrep 313
- commande AT
  - moniteur d'alertes de réplification 237, 238
  - programme Apply 237, 238
  - Programme Capture 237, 238
- commande AT NetView
  - Apply z/OS 238
  - Capture z/OS 238
- commande backup database 30
- commande CHGDPRCAPA 355
- commande CHGJRN 43
- commande CRTDPRTBL 359
- commande CRTJRN 39

- commande CRTJRNRCV 39
- commande DSPJRN 251
- commande ENDDPRAPY 360
- commande ENDDPRCAP 137, 363
- commande GRTPRAUT
  - concession de privilèges sur les modules SQL 33
  - syntaxe 365
- commande GRTOBJAUT 33
- commande INZDPRCAP 374
- commande MODIFY 169, 171
- commande OVRDPRCAPA 375
- commande RCVJRNE 40
- commande RMVDPRREG 380
- commande RMVDPRSUB 381
- commande RMVDPRSUBM 383
- commande RVKDPRAUT 384
- commande SBMJOB 238
- commande STRDPRAPY 144, 386
- commande STRDPRCAP 393
- commande STRJRNPf 39
- commande update database
  - configuration 30
- commande WRKDPRTTRC 401
- commande WRKJOB 250
- commande WRKSBMJOB 250
- commande WRKSBSJOB 250
- commandes
  - asnacmd 277
  - asnapply 270
  - asncap 257
  - asnccmd 266
- commandes d'identification des incidents
  - WRKDPRTTRC 401
- commandes de publication d'événements
  - asnpwd 281
  - asnscrt 285
  - asnsdrop 288
  - asnslst 289
  - asntdiff 229, 290, 294, 302
  - asntrc 305
  - asntrep 313
- commandes de réplication
  - \$TA JES2
    - Apply z/OS 238
    - Capture z/OS 238
  - ADDJOBSCDE 238
  - asnslst 289
  - asntdiff 290, 294, 302
  - asntrep 313
  - AT 237, 238
  - AT NetView
    - Apply z/OS 238
    - Capture z/OS 238
  - backup database 30
  - CRTJRNRCV 39
  - DSPJRN 251
  - System i
    - ADDDPRREG 317
    - ADDDPRSUB 326
    - ADDDPRSUBM 341
    - ANZDPR 352
    - ANZDPRJRN 44
    - CHGDPRCAPA 355
    - CHGJRN 43
    - CRTDPRTBL 359
    - CRTJRN 39

- commandes de réplication (*suite*)
  - System i (*suite*)
    - ENDDPRAPY 360
    - ENDDPRCAP 137, 363
    - GRTPRAUT 33, 365
    - GRTOBJAUT 33
    - INZDPRCAP 374
    - OVRDPRCAPA 375
    - RCVJRNE 40
    - RMVDPRREG 380
    - RMVDPRSUB 381
    - RMVDPRSUBM 383
    - RVKDPRAUT 384
    - SBMJOB 238
    - STRDPRAPY 144, 386
    - STRDPRCAP 393
    - STRJRNPf 39
    - WRKDPRTTRC 401
    - WRKJOB 250
    - WRKSBMJOB 250
    - WRKSBSJOB 250
  - UNIX
    - asnanalyze 278
  - update database configuration 30
  - Windows
    - asnanalyze 278
  - z/OS
    - MODIFY 169
- commandes de réplication Q
  - asnscrt 285
  - asnsdrop 288
  - asnslst 289
  - asnspwd 281
  - asnstrc 305
  - asntdiff 229, 290, 294, 302
  - asntrep 313
- commandes de réplication SQL
  - asnspwd 281
  - asnscrt 285
  - asnsdrop 288
  - asntrc 305
- commandes système
  - asnscrt 285
  - asnsdrop 288
  - asnslst 289
- communication entre composants de la réplication SQL 243
- communication entre les composants de la réplication SQL 243
- comportement d'accès CURRENTLY
  - COMMITTED 167
  - CONCURRENT\_ACCESS
    - \_RESOLUTION 167
- configuration
  - Apply, programmes
    - Linux 29
    - UNIX 29
    - Windows 29
  - connectivité
    - réplication SQL 24
  - journaux 38
  - programme Apply
    - Linux 31
    - UNIX 31
    - Windows 31
  - Programme Capture
    - UNIX 30

- configuration (*suite*)
  - Programme Capture (*suite*)
    - Windows 30
  - programmes Capture
    - Linux 29
    - UNIX 29
    - Windows 29
- configuration d'une réplication à trois niveaux 91
- conflits
  - prévention 8
- connectivité
  - entre des systèmes d'exploitation
    - DB2 24
  - reprise, tables de contrôle 220
- connexion
  - à un serveur System i 24
- conseils et astuces
  - estimation de l'utilisation de l'espace 3
  - suppression de lignes de la table récapitulative Apply 146
  - utilisation de procédures stockées pour les traitements supplémentaires sur les ensembles 157
  - utilisation des paramètres sleep ou des paramètres copyonce 146
  - utilisation des procédures stockées avec ASNDONE 158
  - vérification du résultat du traitement d'un ensemble 146
  - vérification que la capture des modifications a commencé 117
- consignation
  - configuration requise
    - serveurs cible 4
    - serveurs source DB2 3
    - serveurs source relationnels non DB2 9
- console MVS 169, 171
- contrôle
  - démarrage à froid 87, 487
  - données source 51
  - écart dans les données 87, 487
- coordination d'événements de réplication 201
- copie avec régénération intégrale
  - Apply pour System i 389
  - Apply System i 63, 386
  - option d'enregistrement 50
- copie de configurations de réplication 208
- cycle d'abonnement 74

## D

- DB2 Extensions
  - restrictions 104
- DB2 for z/OS
  - planification 12
- DB2 Query Patroller et réplication 38
- DBHEAP, paramètre de configuration de base de données
  - préparation 30
- déclencheurs
  - capture de données 9
  - fusion 9

- déclencheurs de capture
    - autorisations nécessaires 19
    - communication avec
      - Centre de réplication 243
      - programme Apply 243, 246
    - conflits avec des déclencheurs préexistants 9
    - noms des 9
    - planification 9
  - déclencheurs Sybase, modifications nécessaires 10
  - définition de sous-ensembles
    - colonnes dans la cible 96
    - colonnes enregistrées 49
    - données source
      - utilisation de vues 110
    - lignes de modifications dans la cible 97
    - lignes de modifications enregistrées 50
    - techniques avancées
      - pendant l'enregistrement 110
      - utilisation de prédicats 111
  - définition de sous-ensembles verticaux (colonne)
    - dans la cible 96
    - sur la source 49
  - délimiteur # 253
  - délimiteur ; 253
  - délimiteurs
    - scripts SQL générés 253
  - démarrage
    - moniteur d'alertes de réplication
      - UNIX 495
      - Windows 495
    - programme Apply
      - System i 143, 386
      - UNIX 141, 495
      - Windows 141, 495
      - z/OS 141, 171
    - Programme Capture
      - System i 120, 393
      - UNIX 117, 495
      - Windows 117, 495
      - z/OS 117
  - démarrage à chaud
    - Programme Capture
      - System i 393
      - UNIX 122
      - Windows 122
      - z/OS 122
  - démarrage à chaud, programme Capture pour System i 394
  - démarrage à froid, programme Capture pour System i 394
  - démarrages à froid
    - prévention 220
    - Programme Capture
      - System i 393
      - UNIX 122
      - Windows 122
      - z/OS 122
  - dépendances de lecture 60
  - désactivation
    - ensembles d'abonnements 74, 198
    - objets enregistrés 181
  - détection d'écart 87, 487
  - détection de conflit
    - configuration requise 51
    - niveaux 60
    - planification 8
    - présentation 60
    - réplication bidirectionnelle 8
    - réplication entre homologues 8
  - dictionnaires de compression
    - z/OS 214
  - division
    - ensembles d'abonnements 191
  - données
    - définition de sous-ensembles
      - pendant l'enregistrement 110
      - utilisation de prédicats 111
      - utilisation de vues 110
      - utilisation de vues pour indiquer des prédicats 111
    - manipulation 113
    - prévention des suppressions
      - doubles 65
    - récupération
      - tables source 221
    - techniques avancées de définition de sous-ensembles 109
    - transformation
      - à l'abonnement 113
      - à l'enregistrement 113
      - changement de nom de colonnes 98, 115
      - création de colonnes calculées 115
  - données historisées
    - données source 51
    - tables CCD 87, 487
  - données spatiales
    - types 103
  - DPR, enregistrements
    - ajout 317
    - suppression 380
  - droits DBADM (administration de base de données)
    - autorisation 15
  - droits SYSADM (administration de système)
    - autorisations nécessaires 15
- E**
- E-S
    - reprise après erreur 220
  - échecs d'intégrité référentielle 156
  - éditeurs
    - scripts SQL 253
  - élagage
    - table IBMSNAP\_APPLYTRACE 219
    - table IBMSNAP\_APPLYTRAIL 219
    - Table IBMSNAP\_CAPMON 221
    - Table IBMSNAP\_CAPTRACE 221
    - Table IBMSNAP\_UOW 451
    - table SIGNAL 222
    - table UOW (unité de travail) 218
    - Tables CD (change-data) 218
    - tables de contrôle 219
  - élagage automatique 217
  - enregistrement
    - objets 177
  - enregistrement (*suite*)
    - options applicables aux sources
      - arrêt de Capture après une erreur 54
    - colonnes d'image après 51
    - colonnes image-avant 51
    - copie avec régénération
      - intégrale 50
    - détection de conflit 60
    - établissement d'un sous-ensemble (horizontal) de lignes 50
    - établissement d'un sous-ensemble (vertical) de colonnes 49
    - mise à jour sous forme de suppressions et d'insertions 55
    - numéro relatif
      - d'enregistrement 63
    - préfixe image avant 54
    - recapture des modifications (réplication bidirectionnelle) 56
    - réplication en mode capture des modifications 50
    - utilisation des journaux
      - distants 62
    - sources de données IMS 45
    - sources de données relationnelles non-DB2 47
    - tables 177
    - tables DB2 45
    - vues
      - présentation 64, 67
      - procédure 177
  - enregistrement global 437, 438
  - enregistrements
    - ajout 317
    - ajout de colonnes 178
    - arrêt de la capture des modifications 181
    - attributs, modification 178
    - désactivation 181
    - réactivation 182
    - suppression 183, 380
  - enregistrements de journal
    - archivé avant capture 3
    - conservation 212
    - dictionnaires de compression (z/OS) 214
    - gestion 211
    - partitions de base de données multiples 212
  - ensemble d'abonnements, membres
    - ajout 81, 186
    - application d'un sous-ensemble de colonnes 96
    - application d'un sous-ensemble de lignes 97
    - désactivation 187
    - mappage de types de données 98
    - mappages de colonnes 98
    - réplication bidirectionnelle 94
    - réplication multi-niveaux 91
  - ensembles d'abonnements
    - activation de membres 188
    - ajout 326
    - ajout de membres 81, 186
    - cohérence des données 94
    - colonnes 96

- ensembles d'abonnements (*suite*)
  - création 71, 186
  - désactivation 198
  - désactivation de membres 187
  - division 191
  - fusion 194
  - instructions de traitement
    - d'exécution 114
  - instructions SQL 78
  - intégrité référentielle 94
  - lignes 97
  - mini-cycles 74
  - mode de traitement 77
  - modification
    - attributs 188
    - noms 189
    - qualificatifs Apply 196
  - niveau d'activation 74
  - nombre de qualificatifs Apply 70
  - planification
    - fondée sur l'horaire 78
    - fondée sur un événement 78
  - procédures stockées 78
  - réplication bidirectionnelle 94
  - réplication multi-niveaux 91
  - suppression 200, 381
- ensembles d'abonnements inactifs 74
- environnements de réplication
  - copie 208
- espace disque
  - configuration requise 3
- établissement d'un sous-ensemble (horizontal) de lignes
  - dans la cible 97
  - sur la source 50
- établissement d'un sous-ensemble (vertical) de colonnes
  - dans la cible 96
  - sur la source 49
- événements
  - coordination 201

## F

- facteur de groupe 74
- fichier asndone.smp 157
- fichier asnload.ini 164
- fichier de mots de passe 20
  - création 281, 305
- fichier journal Capture 122
- fichiers
  - fichier asndone.smp 157
  - fichier asnload.ini 164
  - fichiers \*.APP.log 146
  - fichiers \*.CAP.log 122
  - fichiers \*.err 146
  - fichiers \*.sqs 146
- fichiers auxiliaires
  - mémoire pour Apply 7
  - mémoire pour Capture 6
- fichiers de diagnostic
  - mémoire 6, 7, 8
- fichiers ERR 146
- fichiers journaux
  - planification de l'impact sur 9
- fichiers journaux APP 146
- fichiers journaux CAP 122

- fichiers SQL
  - édition 253
- fichiers SQS
  - paramètres d'exploitation Apply 146
- files d'attente de messages
  - journaux 43
- files d'attente de messages de journal 43
- fonction load from cursor 163
- fonction trace
  - System i 401
- fragmentation
  - horizontale
    - dans la cible 97
    - sur la source 50
  - réplication bidirectionnelle 8
  - réplication entre homologues 8
  - verticale
    - dans la cible 96
    - sur la source 49
- fusion
  - déclencheurs 9
  - ensembles d'abonnements 194

## G

- gestion du journal de modification du système 40
- Gestionnaire de contrôle de service (SCM)
  - affichage des services de
    - réplication 241
  - arrêt d'un service de réplication 241
  - création de services de
    - réplication 240
  - démarrage d'un service de
    - réplication 241
  - description 239
  - suppression de services de
    - réplication 241
- Gestionnaire de contrôle de service (SCM) Windows
  - commande asnlist 289
  - description 239
  - liste des services de réplication 289
- groupes IASP 27
- groupes IASP (Independent Auxiliary Storage Pool) 27

## I

- IBMQREP\_COLVERSION control
  - table 425
- ID de corrélation 65
- ID utilisateur
  - autorisation 17
  - pour le programme Apply 18
  - pour le programme Capture 17
  - pour les déclencheurs Capture 19
  - stockage 20
- IMS DataPropagator 45
- index
  - clés cible 99
- index cible 99
- instructions SQL
  - définition pour l'ensemble d'abonnements 78

- instructions SQL (*suite*)
  - traitement d'exécution 114
- intégrité référentielle
  - cibles en lecture-écriture 94
- interruption
  - Programme Capture
    - UNIX 138
    - Windows 138
    - z/OS 138

## J

- JCL
  - démarrage du moniteur d'alertes de
    - réplication 169
  - démarrage du programme
    - Apply 169
  - démarrage du programme
    - Capture 169
  - par lots, mode 169
- jointures internes en tant que sources 65
- jonctions
  - sources 65
- journaux
  - avec des heures système
    - différentes 41
  - configuration 38
  - création 39
  - création pour les tables source 39
  - démarrage 39
  - enregistrement en tant que
    - sources 45
  - file d'attente des messages par
    - défaut 43
  - gestion 40
  - journal QSQRN 38
  - types d'entrée 497
  - utilisation 38
  - utilisation de la fonction de journal
    - distant 62
- journaux distants
  - heures système différentes 41
  - sources 62
- journaux source
  - gestion 211

## L

- liaison
  - programme Apply
    - Linux 31
    - UNIX 31
    - Windows 31
  - Programme Capture
    - Linux 31
    - UNIX 31
    - Windows 31
- lignes
  - définition dans les tables cible 97
  - définition de sous-ensembles
    - dans la cible 97
    - sur la source 50
  - disponibles pour la réplication 50
  - enregistrement dans les tables
    - source 50
- LOAD from CURSOR 167

- LOB, type de données
  - restrictions de la réplication
    - bidirectionnelle 94
- logbufsz, paramètre de configuration de base de données
  - Capture 30
- logfilsiz, paramètre de configuration de base de données
  - Capture 30
- logprimary, fichier de configuration de base de données
  - Capture 30
- logsecond, paramètre de configuration
  - Capture 30

## M

- manipulation de données
  - à l'abonnement 113
  - à l'enregistrement 113
  - changement de nom de colonnes 98, 115
  - création de colonnes calculées 115
- mappage
  - colonnes source et colonnes cible 98
  - sources et cibles 81
  - types de données entre différentes tables 98
- marques 508
  - liste de 505
- membres d'un ensemble d'abonnements
  - activation 188
  - actualisation d'un membre 166
  - ajout 341
  - définition d'une clé cible 99
  - nombre par ensemble d'abonnements 70
  - sélection de types cible 84
  - suppression 383
- mémoire
  - configuration requise 3
  - enregistrements 1
  - ensembles d'abonnements 3
  - fichiers auxiliaires d'Apply 7
  - fichiers auxiliaires de Capture 6
  - fichiers de diagnostic 8
  - fichiers de diagnostic d'Apply 7
  - fichiers de diagnostic de Capture 6
  - journal de base de données et données du journal 3
  - lecture des enregistrements de journal 1
  - planification 1
  - programme Apply 3
  - Programme Capture 1
  - table CD 5
  - table UOW 5
  - tables cible 5
  - tables de contrôle 5
  - transactions 1
  - travaux par lots 1
  - utilisation de la table
    - IBMSNAP\_CAPMON pour ajustement 1
  - mentions légales 505
  - Microsoft SQL Server
    - restrictions de réplication 51

- migration
  - planification 1
- mini-cycles 74
- mise en bloc de données 74
- mises à jour
  - conflits 60
  - sous forme de suppressions et d'insertions 55
- mode de partage de données 175
- mode de traitement de table 4, 77
- mode de traitement de transaction 4, 77
- modification des paramètres de Capture
  - System i 355
- modifications de type de données, traitement 180
- modules SQL
  - création pour le programme
    - Apply 33
    - création pour le programme
      - Capture 32
    - création pour Replication
      - Analyzer 33
  - moniteur d'alertes de réplication
    - commande MODIFY 171
    - communication avec
      - Capture 247
      - Centre de réplication 247
      - programme Apply 247
    - planification 237, 238
    - UNIX
      - démarrage 495
      - vérification du statut 249
    - Windows
      - démarrage 495
      - vérification du statut 249
    - z/OS
      - vérification du statut 249
- mots de passe
  - stockage 20

## N

- noms
  - des déclencheurs de capture 9
  - ensembles d'abonnements 189
  - règles de qualificatif Apply 255
  - règles de qualificatif du
    - moniteur 255
  - règles de schémas Capture 255
  - services Windows 255
- non-concordance de temps (System i) 41
- numéro relatif d'enregistrement
  - en tant que clé primaire System i 63
  - prise en charge de System i 63
  - utilisé comme clé cible 99

## O

- objets
  - arrêt de la capture des modifications 181
  - désactivation 181
  - enregistrement 177
  - modification des attributs 178
  - réactivation 182

- opérations ALTER COLUMN SET DATA TYPE 180
- opérations SET DATA TYPE 180
- optimisation
  - paramètre commit\_interval 1
  - paramètre memory\_limit 1
- option d'arrêt de Capture après une erreur 54

## P

- packages
  - redéfinition 216
- pages de codes
  - compatible 11
  - traduction 10
  - variable d'environnement
    - DB2CODEPAGE 11
- paramètre
  - variables d'environnement
    - Capture 29
- paramètre add\_partition
  - présentation 122, 123
- paramètre ALWINACT 386, 390
- paramètre apply\_path 146
- paramètre apply\_qual 146
- paramètre APYQUAL 386, 388
- paramètre asynchlogrd
  - présentation 122
- paramètre autoprune
  - présentation 122
- paramètre autostop 122
- paramètre caf 122, 146
- paramètre CAPTLLIB 393, 396
- paramètre capture\_path 122
- paramètre capture\_schema 122
- paramètre capture\_server 122
- paramètre CLNUPITV 393, 396
- paramètre commit\_interval
  - optimisation 1
  - présentation 122
- paramètre control\_server 146
- paramètre copyonce 146
- paramètre COPYONCE 386, 390
- paramètre CTLSVR 386, 388
- paramètre db2\_subsystem 146
- paramètre de configuration applheapsz
  - exécution du programme capture 30
- paramètre de configuration maxappls
  - Capture 30
- paramètre delay 146
- paramètre DELAY 386, 390
- paramètre errwait 146
- paramètre FRFCFRQ 393, 399
- paramètre FULLREFPGM 386, 389
- paramètre INACTMSG 386, 389
- paramètre inamsq 146
- paramètre JOBID 386, 387, 393, 395
- paramètre JRN 393, 397
- paramètre LAG 393, 398
- paramètre lag\_limit 122
- paramètre loadxit 146
- paramètre logreuse
  - Apply 146
  - Capture 122
- paramètre logstdout
  - Apply 146

- paramètre logstdout (*suite*)
  - Capture 122
- paramètre MEMLMT 393, 398
- paramètre memory\_limit
  - optimisation 1
  - présentation 122
- paramètre monitor\_interval
  - Capture 122
- paramètre monitor\_limit 122
- paramètre MONITV 393, 397
- paramètre MONLMT 393, 397
- paramètre notify 146
- paramètre opt4one
  - paramètres d'exploitation Apply 146
- paramètre OPTSNGSET 386, 391
- paramètre prune\_interval 122
- paramètre pwdfile 146
- paramètre RESTART 393, 394
- paramètre RETAIN 393, 398
- paramètre retention\_limit 122
- paramètre RTYWAIT 386, 390
- paramètre sleep 146
- paramètre sleep\_interval 122
- paramètre spillfile 146
- paramètre sqlerrcontinue 146
- paramètre startmode
  - présentation 122
- paramètre SUBNFYPGM 386, 389
- paramètre term
  - Apply 146
  - Capture 122
- paramètre TRACE 386, 388
- paramètre trace\_limit
  - présentation 122
- paramètre TRCLMT 393, 397
- paramètre trreuse 146
- paramètre TRLREUSE 386, 391
- paramètre USER 386, 387
- paramètre WAIT 393, 395
- paramètres
  - appel
    - analyseur pour System i 352
    - Apply, programme pour System i 143, 386
    - Capture, programme pour System i 355, 393
    - commandes de réplication pour System i 317, 326, 341, 359, 360, 363, 365, 374, 375, 380, 381, 383, 384, 386, 393, 401
    - programme Apply pour UNIX 146
    - programme Apply pour Windows 146
    - programme Apply pour z/OS 146
    - programme Capture pour UNIX 122
    - programme Capture pour Windows 122
    - programme Capture pour z/OS 122
  - paramètres, appel (*suite*)
    - commandes de réplication
      - pour System i 318, 328, 342, 360, 361, 363, 374, 376, 380, 381, 383, 385, 387, 394, 402
    - programme Apply
      - pour System i 387
    - Programme Capture
      - pour System i 355, 394
  - paramètres d'appel
    - Analyseur
      - pour System i 352
      - System i 352
    - commandes de réplication
      - pour System i 318, 328, 342, 360, 361, 363, 374, 376, 380, 381, 383, 385, 387, 394, 402
      - System i 317, 326, 341, 359, 360, 363, 365, 374, 375, 380, 381, 383, 384, 386, 393, 401
    - programme Apply
      - pour System i 387
      - System i 143, 386
      - UNIX 146
      - Windows 146
      - z/OS 146
    - Programme Capture
      - pour System i 355, 394
      - System i 117, 120, 355, 393
      - UNIX 122
      - Windows 122
      - z/OS 122
  - paramètres de configuration
    - APPLHEAPSZ 30
    - DBHEAP 30
    - LOGBUFSZ 30
    - LOGFILSIZ 30
    - LOGPRIMARY 30
    - LOGSECOND 30
    - MAXAPPLS 30
  - partitions de base de données
    - Capture 34
  - partitions de base de données multiples
    - enregistrements de journal 212
  - performances
    - ajustement 12
  - personnalisation
    - scripts SQL 253
  - planification
    - besoins en mémoire 3
    - coexistence de déclencheurs 9
    - débites des transactions 9
    - détection de conflit 8, 60
    - impact du journal 9
    - mémoire 1
    - migration 1
    - verrous sur des tables CCD 10
  - planification fondée sur l'horaire 78
  - planification fondée sur un événement 78
  - plannings
    - ensembles d'abonnements 78
    - programmes de réplication 237
  - plans
    - redéfinition 216
  - points de reprise
    - distribué 203
- points de reprise distribués 203
- prédicats
  - définition de sous-ensembles 111
  - définition pour des tables cible 97
- préfixe image avant 54
- préfixes
  - image avant 54
- prise en charge de nom long
  - planification 12
- procédures CALL
  - définition pour l'ensemble d'abonnements 78
  - traitement avant et après exécution 114
- procédures stockées
  - définition pour l'ensemble d'abonnements 78
  - manipulation des données 114
- programme Apply
  - autorisations nécessaires 18
  - commande MODIFY 171
  - commandes
    - asnacmd 277
    - asnapply 270
    - Réplication SQL 257
  - communication avec
    - Centre de réplication 243
    - déclencheurs de capture 246
    - moniteur d'alertes de réplication 247
    - programme Capture 243
  - connectivité 24
  - définition de valeurs par défaut pour les paramètres 155
  - échecs d'intégrité référentielle 156
  - fonctionnement 141
  - ID utilisateur 18
  - instructions de traitement d'exécution 114
  - Linux
    - configuration 29, 31
    - liaison 31
  - mini-cycles 74
  - mise en bloc de données 74
  - mode de traitement de table 77
  - mode de traitement de transaction 77
  - modification des valeurs de paramètres 155
  - planification 237
  - pour System i
    - paramètre ALWINACT 390
    - paramètre APYQUAL 388
    - paramètre COPYONCE 390
    - paramètre CTLSVR 388
    - paramètre DELAY 390
    - paramètre FULLREFFPGM 389
    - paramètre INACTMSG 389
    - paramètre JOBID 387
    - paramètre OPTSNGSET 391
    - paramètre RTYWAIT 390
    - paramètre SUBNFYPGM 389
    - paramètre TRACE 388
    - paramètre TRLREUSE 391
    - paramètre USER 387
  - System i
    - arrêt 156, 360



programme Apply (*suite*)

System i (*suite*)

- configuration 38
- création de modules SQL 33
- démarrage 143, 386
- paramètre ALWINACT 386
- paramètre APYQUAL 386
- paramètre COPYONCE 386
- paramètre CTLSVR 386
- paramètre DELAY 386
- paramètre FULLREFPGM 386
- paramètre INACTMSG 386
- paramètre JOB 386
- paramètre OPTSNGSET 386
- paramètre RTYWAIT 386
- paramètre SUBNFYPGM 386
- paramètre TRACE 386
- paramètre TREREUSE 386
- paramètre USER 386
- planification 238
- vérification du statut 250

UNIX

- arrêt 156
- configuration 29, 31
- démarrage 141, 495
- liaison 31
- paramètre apply\_path 146
- paramètre apply\_qual 146
- paramètre caf 146
- paramètre control\_server 146
- paramètre copyonce 146
- paramètre delay 146
- paramètre errwait 146
- paramètre inamsg 146
- paramètre loadxit 146
- paramètre logreuse 146
- paramètre logstdout 146
- paramètre notify 146
- paramètre opt4one 146
- paramètre pwdfile 146
- paramètre sleep 146
- paramètre spillfile 146
- paramètre sqlerrcontinue 146
- paramètre term 146
- paramètre trlreuse 146
- paramètres par défaut 144
- vérification du statut 249

Windows

- arrêt 156
- configuration 29, 31
- démarrage 141, 495
- liaison 31
- paramètre apply\_path 146
- paramètre apply\_qual 146
- paramètre caf 146
- paramètre control\_server 146
- paramètre copyonce 146
- paramètre delay 146
- paramètre errwait 146
- paramètre inamsg 146
- paramètre loadxit 146
- paramètre logreuse 146
- paramètre logstdout 146
- paramètre notify 146
- paramètre opt4one 146
- paramètre pwdfile 146
- paramètre sleep 146

programme Apply (*suite*)

Windows (*suite*)

- paramètre spillfile 146
- paramètre sqlerrcontinue 146
- paramètre term 146
- paramètre trlreuse 146
- paramètres par défaut 144
- vérification du statut 249

z/OS

- arrêt 156
- configuration 34
- démarrage 141, 171
- paramètre apply\_path 146
- paramètre apply\_qual 146
- paramètre caf 146
- paramètre control\_server 146
- paramètre copyonce 146
- paramètre db2\_subsystem 146
- paramètre delay 146
- paramètre errwait 146
- paramètre inamsg 146
- paramètre loadxit 146
- paramètre logreuse 146
- paramètre logstdout 146
- paramètre notify 146
- paramètre opt4one 146
- paramètre pwdfile 146
- paramètre sleep 146
- paramètre spillfile 146
- paramètre term 146
- paramètre trlreuse 146
- paramètres 144
- valeur par défaut 144
- vérification du statut 249

programme Capture

- communication avec
  - Centre de réplication 243
  - programme Apply 243

Programme Capture

- autorisations nécessaires 17
- commande MODIFY 171
- commandes 257
  - asnccmd 266
- communication avec
  - moniteur d'alertes de réplication 247
  - programme Apply 244
- connectivité 24
- définition de valeurs par défaut pour les paramètres 133
- définition de variables d'environnement 29
- emplacement de démarrage 122
- exécution de plusieurs 28
- ID utilisateur 17

Linux

- configuration 29
- liaison 31
- mémoire utilisée par 1
- modification des schémas 184
- modification des valeurs de paramètres 133
- modification du comportement en cours de fonctionnement 135
- planification 237
- pour System i
  - paramètre CAPCTLLIB 396

Programme Capture (*suite*)

pour System i (*suite*)

- paramètre CLNUPITV 396
- paramètre FRCFRQ 399
- paramètre JOB 395
- paramètre JRN 397
- paramètre LAG 398
- paramètre MEMLMT 398
- paramètre MONITV 397
- paramètre MONLMT 397
- paramètre RETAIN 398
- paramètre TRCLMT 397
- paramètre WAIT 395
- prévention des démarrages à froid 220
- signaux 201
- System i
  - arrêt 137, 363
  - autorisations nécessaires 15
  - configuration 38
  - création de modules SQL 32
  - démarrage 120, 393
  - fonctionnement 117
  - journaux et récepteurs de journal, gestion 40
  - modification des attributs 355
  - paramètre CAPCTLLIB 393
  - paramètre CLNUPITV 393
  - paramètre FRCFRQ 393
  - paramètre JOB 393
  - paramètre JRN 393
  - paramètre LAG 393
  - paramètre MEMLMT 393
  - paramètre MONITV 393
  - paramètre MONLMT 393
  - paramètre RESTART 393
  - paramètre RETAIN 393
  - paramètre TRCLMT 393
  - paramètre WAIT 393
  - paramètres de démarrage à chaud 393
  - paramètres de démarrage à froid 393
  - paramètres par défaut 120
  - planification 238
  - progression de 251
  - réinitialisation 374
  - substitution des attributs de 375
  - types d'entrée de journal 497
  - vérification du statut 250

UNIX

- arrêt 137
- configuration 29, 30
- démarrage 117, 495
- fonctionnement 117
- interruption 138
- liaison 31
- paramètre add\_partition 122
- paramètre asynchlogrd 122
- paramètre autoprune 122
- paramètre autostop 122
- paramètre caf 122
- paramètre capture\_path 122
- paramètre capture\_schema 122
- paramètre capture\_server 122
- paramètre commit\_interval 122
- paramètre lag\_limit 122

- Programme Capture (*suite*)
    - UNIX (*suite*)
      - paramètre logreuse 122
      - paramètre logstdout 122
      - paramètre memory\_limit 122
      - paramètre monitor\_interval 122
      - paramètre monitor\_limit 122
      - paramètre prune\_interval 122
      - paramètre retention\_limit 122
      - paramètre sleep\_interval 122
      - paramètre startmode 122
      - paramètre term 122
      - paramètre trace\_limit 122
      - paramètres de démarrage à chaud 122
      - paramètres de démarrage à froid 122
      - paramètres par défaut 120
      - réinitialisation 138
      - reprise 139
      - vérification du statut 249
    - Windows
      - arrêt 137
      - configuration 29, 30
      - démarrage 117, 495
      - fonctionnement 117
      - interruption 138
      - liaison 31
      - paramètre add\_partition 122
      - paramètre asynchlogrd 122
      - paramètre autoprune 122
      - paramètre autostop 122
      - paramètre caf 122
      - paramètre capture\_path 122
      - paramètre capture\_schema 122
      - paramètre capture\_server 122
      - paramètre commit\_interval 122
      - paramètre lag\_limit 122
      - paramètre logreuse 122
      - paramètre logstdout 122
      - paramètre memory\_limit 122
      - paramètre monitor\_interval 122
      - paramètre monitor\_limit 122
      - paramètre prune\_interval 122
      - paramètre retention\_limit 122
      - paramètre sleep\_interval 122
      - paramètre startmode 122
      - paramètre term 122
      - paramètre trace\_limit 122
      - paramètres de démarrage à chaud 122
      - paramètres de démarrage à froid 122
      - paramètres par défaut 120
      - réinitialisation 138
      - reprise 139
      - vérification du statut 249
  - z/OS
    - arrêt 137
    - configuration 34
    - démarrage 117
    - fonctionnement 117
    - interruption 138
    - paramètre add\_partition 122
    - paramètre asynchlogrd 122
    - paramètre autoprune 122
    - paramètre autostop 122
- Programme Capture (*suite*)
  - z/OS (*suite*)
    - paramètre caf 122
    - paramètre capture\_path 122
    - paramètre capture\_schema 122
    - paramètre capture\_server 122
    - paramètre commit\_interval 122
    - paramètre lag\_limit 122
    - paramètre logreuse 122
    - paramètre logstdout 122
    - paramètre memory\_limit 122
    - paramètre monitor\_interval 122
    - paramètre monitor\_limit 122
    - paramètre prune\_interval 122
    - paramètre retention\_limit 122
    - paramètre sleep\_interval 122
    - paramètre startmode 122
    - paramètre term 122
    - paramètre trace\_limit 122
    - paramètres de démarrage à chaud 122
    - paramètres de démarrage à froid 122
    - paramètres par défaut 120
    - réinitialisation 138
    - reprise 139
    - vérification du statut 249
  - promotion
    - configurations de réplication 208
  - pseudonymes
    - enregistrement
      - tables relationnelles non DB2 47
    - fonction load from cursor 163
    - restrictions
      - avec les tables CCD 51
      - réplication bidirectionnelle 56, 94
      - réplication multi-niveaux 91
      - tables d'agrégation 86
  - publication d'événement
    - stockage des ID utilisateur et des mots de passe 20
- Q**
  - QTIME, différences de valeur 41
  - qualificatifs Apply
    - modification dans des ensembles d'abonnements 196
    - nombre d'ensembles d'abonnements associés 70
    - règles d'attribution de nom 255
    - utiliser au démarrage du programme Apply
      - Linux, UNIX et Windows 141
      - System i 143
  - qualificatifs de moniteur
    - règles d'attribution de nom 255
- R**
  - rapports de l'analyseur
    - commande ANZDPR 352
    - commande asnanalyze 278
  - réactivation
    - enregistrements 182
    - objets 182
  - réactivation (*suite*)
    - tables 182
  - recapture
    - modifications bidirectionnelles 56
  - récepteurs de journal
    - accès 213
    - actuel, taille 3
    - création pour les tables source 39
    - gestion 40, 211
    - gestion de système 40
    - gestion des utilisateurs 43
    - routine d'exit supprimer un récepteur de journal 44
    - seuil 40
  - récupération aval
    - Programme Capture 30
  - redéfinition
    - packages
      - Linux, UNIX, Windows et z/OS 216
  - régénération intégrale
    - System i, problèmes dus à la non-concordance des temps 41
  - règles de capture de lignes
    - réplication en mode capture des modifications 50
  - réinitialisation
    - Programme Capture
      - UNIX 138
      - Windows 138
      - z/OS 138
  - rendement
    - débits
      - déclencheurs de capture 9
  - réorganisation
    - tables de contrôle 216
  - réplication, commandes de
    - asntdiff 229
  - Replication Analyzer
    - pour System i
      - paramètres d'appel 352
    - System i
      - création de modules SQL 33
      - paramètres d'appel 352
  - réplication bidirectionnelle
    - définition d'ensembles d'abonnements 94
    - détection de conflit
      - configuration requise 51, 60
      - planification pour 8
      - présentation 60
    - fragmentation pour 8
    - recapture des modifications 56
  - réplication DB2
    - autorisations nécessaires 15
  - réplication en mode capture des modifications
    - description 50
    - option d'enregistrement 50
  - réplication en mode régénération différentielle 50
  - réplication entre homologues
    - détection de conflit 8
  - réplication hétérogène
    - enregistrement de sources 47
    - restrictions
      - réplication bidirectionnelle 56, 94

- réplication hétérogène (*suite*)
    - restrictions (*suite*)
      - réplication multi-niveaux 91
      - tables CCD 51
      - tables d'agrégation 86
  - réplication multi-niveaux
    - définition d'ensembles
    - d'abonnements 91
  - réplication Q
    - stockage des ID utilisateur et des mots de passe 20
  - réplication SQL
    - déclencheurs Sybase 10
    - présentation de la planification 1
    - vérification du statut 249
  - réplication transférée
    - présentation 91
  - reprise
    - Programme Capture
      - UNIX 139
      - Windows 139
      - z/OS 139
  - réseaux
    - connectivité 24
  - restrictions
    - chiffrement de données 103
    - clause WHERE 97
    - colonnes LONG de tables Oracle 103
    - Microsoft SQL Server 51
    - noms de colonnes, limites 51
    - objets LOB de DB2 Extensions 104
    - procédures stockées 114
    - réplication hétérogène 51, 91, 94
    - sources de données relationnelles non-DB2 56, 60
    - sources Oracle 103
    - Sybase 51
    - tables ASCII 493
    - tables CCD 94
    - tables cible existantes 96
    - tables Unicode 493
    - type de données spatiales 103
    - types de données abstraits 103
    - types de données définis par l'utilisateur 103
    - types de données distincts 103
    - types de données LOB 94
    - types de données LONG VARCHAR 103
    - types de données LONG VARGRAPHIC 103
    - vues 67
  - restrictions sur le chiffrement des données 103
  - routine d'exit ASNDONE
    - transactions rejetées 60
    - utilisation
      - Linux, UNIX et Windows 157
      - System i 158
  - routine d'exit ASNLOAD
    - description 160
    - personnalisation du comportement 163
    - System i 165
    - traitement d'erreurs 160
    - UNIX 160
  - routine d'exit supprimer un récepteur de journal
    - a propos de 44
  - routines d'exit
    - ASNDONE
      - utilisation 157, 158
    - ASNLOAD
      - personnalisation 163
      - System i 165
      - UNIX 160
      - utilisation 160
      - Windows 160
      - z/OS 162
  - supprimer un récepteur de journal (System i) 44
  - RRN 63
- S**
- schémas
    - modification 184
    - règles d'attribution de nom 255
  - schémas de Capture
    - modification 184
    - règles d'attribution de nom 255
    - utilisation de plusieurs 28
  - SCM (Service Control Manager)
    - affichage des services de
      - réplication 241
    - arrêt de services de réplication 241
    - création de services de
      - réplication 240
    - démarrage de services de
      - réplication 241
    - description 239
    - suppression de services de
      - réplication 241
  - scripts SQL
    - personnalisation 253
  - scripts SQL générés 253
  - serveur de contrôle Capture
    - plusieurs schémas de Capture 28
  - serveur Q Capture
    - IBMQREP\_COLVERSION control table 425
    - table de contrôle
      - IBMQREP\_IGNTRAN 430
    - table de contrôle
      - IBMQREP\_IGNTRANTRC 430
    - Table de contrôle
      - IBMQREP\_TABVERSION 450
  - serveurs cible
    - impact du journal 4
  - serveurs relationnels non DB2
    - connexion 25
  - serveurs source
    - DB2
      - impact du journal 3
  - serveurs source (*suite*)
    - données relationnelles non DB2
      - impact du journal 9
  - service
    - clients 501
  - service Windows
    - création 285, 288
  - services de réplication
    - affichage 241
    - arrêt 241
    - création 240
    - démarrage 241
    - description 239
    - liste 289
    - nom 239
    - nom d'affichage 239
    - suppression 241
  - Services logiciels
    - contact 501
  - signaux
    - CAPSTART 205
    - CAPSTOP 206
    - définition de points de reprise distribués 203
    - STOP 202, 203
    - USER 201
  - signaux CAPSTART 205
  - signaux CAPSTOP 206
  - signaux Capture 201
  - signaux STOP 202, 203
  - signaux USER 201
  - sites Web
    - non IBM 503
  - sources
    - abonnement à 71
    - enregistrement
      - données relationnelles non DB2 47
      - sources de données IMS 45
      - tables DB2 45
      - vues 64, 67
    - enregistrement de colonnes 49
    - gestion des tables CCD 67
    - lignes d'enregistrement 50
    - mappage vers les cibles 81
    - options d'enregistrement
      - arrêt de Capture après une erreur 54
      - colonnes d'image après 51
      - colonnes image-avant 51
      - copie avec régénération intégrale 50
      - détection de conflit 60
      - établissement d'un sous-ensemble (horizontal) de lignes 50
      - établissement d'un sous-ensemble (vertical) de colonnes 49
      - mise à jour sous forme de suppressions et d'insertions 55
      - numéro relatif
        - d'enregistrement 63
      - préfixe image avant 54
      - recapture des modifications (réplication bidirectionnelle) 56
      - réplication en mode capture des modifications 50

- sources (*suite*)
    - options d'enregistrement (*suite*)
      - utilisation des journaux distants 62
    - tables de modification cohérente des données (CCD) 91
  - sources de données IMS
    - enregistrement 45
    - gestion des tables CCD 67
    - utilisation de tables CCD 45
  - sources de données non relationnelles
    - gestion des tables CCD 67
    - utilisation de tables CCD 45
  - sources de données relationnelles non-DB2
    - enregistrement 47
    - restrictions
      - réplication bidirectionnelle 56, 60, 94
      - réplication multi-niveaux 91
      - tables d'agrégation 86
    - serveurs source 9
    - utilisation de tables CCD 47
    - verrous 10
  - sources de données System i
    - avec une journalisation distante 62
    - intégrité référentielle 63
  - sources de réplication
    - abonnement à 71
    - enregistrement
      - colonnes 49
      - lignes 50
      - sources de données IMS 45
      - sources de données relationnelles non-DB2 47
      - tables DB2 45
      - vues 67
    - gestion des tables CCD 67
    - jonctions 65
    - mappage vers les cibles 81
    - tables de modification cohérente des données (CCD) 91
  - statut
    - Capture, programme 249
    - moniteur d'alertes de réplication 249
    - programme Apply 249, 250
    - Programme Capture 250
    - travaux de journal 250
  - substitution des attributs
    - Programme Capture 375
  - support de langue nationale
    - pages de code pour la réplication SQL 11
  - suppressions doubles 65
  - supprimer le programme d'exit d'un récepteur de journal 213
  - surveillance
    - statut des programmes 250
    - System i 251
  - Sybase
    - restrictions de réplication 51
  - synchronisation
    - utilitaires asntdiff et asntrepare 223
  - synchronisation temporelle 78
  - System i
    - serveurs
      - connexion 24
  - système, commandes
    - asnacmd 277
    - asnapply 270
    - asncap 257
    - asnccmd 266
    - asnpwd 281
    - asntdiff 229, 290, 294, 302
    - asntrc 305
    - asntrepare 313
  - systèmes source
    - gestion 211
- T**
- table APPPARMS (paramètres Apply)
    - modification 155
  - table ASN.IBMSNAP\_FEEDETL 468
  - Table ASN.IBMSNAP\_FEEDETL 468
  - table CAPPARMS (paramètres Capture)
    - modification 136
    - utilisation 133
  - table d'ensembles d'abonnements (SUBS\_MEMBR) 163, 472
  - table d'événements d'abonnement (SUBS\_EVENT)
    - affichage d'événements 78
  - table de contrôle
    - IBMQREP\_IGNTRAN 430
  - table de contrôle
    - IBMQREP\_IGNTRANTRC 430
  - Table de contrôle
    - IBMQREP\_TABVERSION 450
  - table de copie utilisateur
    - définition 84
    - structure 490
    - utilisation 85
  - table des différences 223
  - table des paramètres Apply
    - modification 155
  - table des paramètres Capture (CAPPARMS)
    - modification 136
    - utilisation 133
  - table des unités de travail (UOW)
    - besoins en mémoire 5
    - colonnes de tables CCD 87, 487
    - élagage 218
  - table IBMSNAP\_APPENQ 454
  - table IBMSNAP\_APPLEVEL
    - structure 455
  - table IBMSNAP\_APPLY\_JOB 455
  - table IBMSNAP\_APPLYMON 456
  - table IBMSNAP\_APPLYTRACE
    - élagage 219
    - structure 461
  - table IBMSNAP\_APPLYTRAIL
    - élagage 219
    - structure 462
  - table IBMSNAP\_APPPARMS 457
    - utilisation 155
  - table IBMSNAP\_AUTHTKN 416
  - table IBMSNAP\_CAPENQ 417
  - table IBMSNAP\_CAPMON
    - structure 418
  - Table IBMSNAP\_CAPMON
    - élagage 221
  - Table IBMSNAP\_CAPPARMS
    - structure 420
  - Table IBMSNAP\_CAPSCHEMAS 424
  - Table IBMSNAP\_CAPTRACE
    - élagage 221
    - structure 426
  - table IBMSNAP\_PARTITIONINFO 431
  - Table IBMSNAP\_PARTITIONINFO 431
  - table IBMSNAP\_PRUNCNTL 432
  - Table IBMSNAP\_PRUNE\_LOCK 434
  - Table IBMSNAP\_PRUNE\_SET 435
  - table IBMSNAP\_REG\_EXT 435
  - Table IBMSNAP\_REG\_SYNCH 444
  - Table IBMSNAP\_REGISTER 437
  - Table IBMSNAP\_RESTART 445
  - Table IBMSNAP\_SEQTABLE 447
  - table IBMSNAP\_SIGNAL
    - structure 447
  - Table IBMSNAP\_SUBS\_COLS 469
  - Table IBMSNAP\_SUBS\_EVENT
    - structure 471
  - Table IBMSNAP\_SUBS\_MEMBR 166
  - table IBMSNAP\_SUBS\_SET 477
  - Table IBMSNAP\_SUBS\_SET 477
  - Table IBMSNAP\_SUBS\_STMTS 483
  - Table IBMSNAP\_UOW
    - élagage 451
    - structure 451
  - table SIGNAL
    - élagage 222
  - table SUBS\_EVENT (événements d'abonnement)
    - affichage d'événements 78
  - table SUBS\_MEMBR (membres d'ensembles d'abonnements) 163, 472
  - table SYSIBM.SYSCOLUMNS 23
  - Table SYSIBM.SYSTABLEPART 23
  - table SYSIBM.SYSTABLES 23
  - table UOW (unité de travail)
    - besoins en mémoire 5
    - colonnes de tables CCD 87, 487
    - élagage 218
  - tables
    - agrégation de base 485
    - agrégation des modifications 486
    - ajout de colonnes 178
    - arrêt de la capture des modifications 181
    - ASN.IBMSNAP\_FEEDETL 468
    - AUTHTKN (référence croisée de qualificatif Apply) 416
    - CCD (consistent-change-data)
      - serveur de contrôle Capture 427
    - chargement 166
    - cibles 96
    - copie utilisateur 490
    - désactivation 181
    - détection de conflit pour 8
    - enregistrement
      - DB2 45
      - données relationnelles non DB2 47
      - procédure 177
      - gestion des tables CCD 67
      - IBMQREP\_TABVERSION 425
      - IBMQREP\_IGNTRAN 430
      - IBMQREP\_IGNTRANTRC 430

tables (*suite*)

- IBMQREP\_TABVERSION 450
- IBMSNAP\_APPENQ 454
- IBMSNAP\_APPLEVEL 455
- IBMSNAP\_APPLY\_JOB 455
- IBMSNAP\_APPLYMON 456
- IBMSNAP\_APPLYTRACE 461
- IBMSNAP\_APPLYTRAIL 462
- IBMSNAP\_APPPARMS 457
- IBMSNAP\_CAPENQ 417
- IBMSNAP\_CAPMON 221, 418
- IBMSNAP\_CAPPARMS 420
- IBMSNAP\_CAPSCHEMAS 424
- IBMSNAP\_CAPTRACE 221, 426
- IBMSNAP\_PARTITIONINFO 431
- IBMSNAP\_PRUNCNTL 432
- IBMSNAP\_PRUNE\_LOCK 434
- IBMSNAP\_PRUNE\_SET 435
- IBMSNAP\_REG\_EXT 435
- IBMSNAP\_REG\_SYNCH 444
- IBMSNAP\_REGISTER 437
- IBMSNAP\_RESTART 445
- IBMSNAP\_SEQTABLE 447
- IBMSNAP\_SIGNAL 447
- IBMSNAP\_SUBS\_COLS 469
- IBMSNAP\_SUBS\_EVENT 471
- IBMSNAP\_SUBS\_SET 477
- IBMSNAP\_SUBS\_STMTS 483
- IBMSNAP\_UOW 451
- modification des attributs 178
- modification des données 429
- point de cohérence 489
- réactivation 182
- réplique 8, 490
- SUBS\_MEMBR (membres d'ensembles d'abonnements) 163, 472
- suppression des enregistrements 183
- tables cible 222
  - gestion 222
- tables de contrôle
  - dynamiques 215
  - élagage 219
  - gestion 215
  - réorganisation 216
  - reprise après échec de connectivité 220
  - reprise après erreur E-S 220
  - statiques 216
  - utilitaire RUNSTATS 215
- tables ASCII 493
- tables CCD externes
  - réplication multi-niveaux 91
- tables CCD internes
  - cibles multiples 89
- tables CD (change-data)
  - structure 429
- Tables CD (change-data)
  - besoins en mémoire 5
  - élagage 218
  - jonction 65
  - pour les vues 64
  - récapitulatif de contenu 86
- tables cible
  - agrégation de base
    - définition 84
    - structure 485
    - utilisation 86

tables cible (*suite*)

- agrégation des modifications
  - définition 84
  - structure 486
  - utilisation 86
- application d'un sous-ensemble de colonnes 96
- application d'un sous-ensemble de lignes 97
- besoins en mémoire 5
- CCD (consistent-change-data)
  - présentation 84
- copie utilisateur
  - définition 84
  - structure 490
  - utilisation 85
- définies par l'utilisateur 84, 96
- définition d'une clé cible 99
- définition de colonnes 96
- définition de lignes 97
- fragmentation 96, 97
- gestion 222
- mappage vers les sources 81
- multiples 89
- nouvelles colonnes pour 115
- point de cohérence
  - définition 84
  - structure 489
  - utilisation 85
- réparation 232
- réplique
  - définition 84
  - détection de conflit pour 8
  - structure 490
  - utilisation 94
- structures de table, guide de référence 485
- tables d'agrégation
  - agrégation de base 86, 485
  - agrégation des modifications 86, 486
- tables d'agrégation de base
  - définition 84
  - structure 485
  - utilisation 86
- tables d'agrégation des modifications
  - définition 84
  - structure 486
  - utilisation 86
- tables DB2
  - enregistrement 45
- tables de contrôle
  - ASN.IBMSNAP\_FEEDETL 468
  - autorisations nécessaires System i 38
  - besoins en mémoire 5
  - CCD (consistent-change-data)
    - serveur de contrôle Capture 427
  - création 26
    - dans des groupes IASP 27
    - plusieurs ensembles 28
    - plusieurs partitions de base de données 29
    - pour des sources relationnelles non DB2 27
    - sur System i 27, 359
    - système d'exploitation avec plusieurs bases de données 26
  - dynamiques 215

tables de contrôle (*suite*)

- élagage 219
- gestion 215
- guide de référence
  - aperçu rapide 407
  - Serveur Capture 414
  - serveur cible 485
  - serveur de contrôle Apply 454
- IBMSNAP\_APPENQ 454
- IBMSNAP\_APPLEVEL 455
- IBMSNAP\_APPLY\_JOB 455
- IBMSNAP\_APPLYMON 456
- IBMSNAP\_APPLYTRACE 461
- IBMSNAP\_APPLYTRAIL 462
- IBMSNAP\_APPPARMS 457
- IBMSNAP\_AUTHTKN 416
- IBMSNAP\_CAPENQ 417
- IBMSNAP\_CAPMON
  - élagage 221
  - structure 418
- IBMSNAP\_CAPPARMS
  - structure 420
- IBMSNAP\_CAPSCHEMAS 424
- IBMSNAP\_CAPTRACE
  - élagage 221
  - structure 426
- IBMSNAP\_PARTITIONINFO 431
- IBMSNAP\_PRUNCNTL 432
- IBMSNAP\_PRUNE\_LOCK 434
- IBMSNAP\_PRUNE\_SET 435
- IBMSNAP\_REG\_EXT 435
- IBMSNAP\_REG\_SYNCH 444
- IBMSNAP\_REGISTER 437
- IBMSNAP\_RESTART 445
- IBMSNAP\_SEQTABLE 447
- IBMSNAP\_SIGNAL 447
- IBMSNAP\_SUBS\_COLS 469
- IBMSNAP\_SUBS\_EVENT 471
- IBMSNAP\_SUBS\_SET 477
- IBMSNAP\_SUBS\_STMTS 483
- IBMSNAP\_UOW 451
- modification des données 429
- octroi des droits System i 15, 365
- redéfinition, modules et plans 216
- réorganisation 216
- reprise après échec de connectivité 220
- reprise après erreur E-S 220
- révocation des droits d'accès System i 384
- Serveur Capture 414
- serveur cible 485
- serveur de contrôle Apply 454
- serveur Q Capture
  - IBMQREP\_COLVERSION 425
  - IBMQREP\_IGNTRAN 430
  - IBMQREP\_IGNTRANTRC 430
  - IBMQREP\_TABVERSION 450
- statiques 216
- SUBS\_MEMBR (membres d'ensembles d'abonnements) 472
- utilitaire RUNSTATS 215
- tables de contrôle Apply
  - APPPARMS (paramètres Apply)
    - modification 155
  - ASN.IBMSNAP\_FEEDETL 468
  - IBMSNAP\_APPENQ 454

- tables de contrôle Apply *(suite)*
    - IBMSNAP\_APPLY\_JOB 455
    - IBMSNAP\_APPLYMON 456
    - IBMSNAP\_APPLYTRACE 461
    - IBMSNAP\_APPLYTRAIL 462
    - IBMSNAP\_APPPARMS 457
      - utilisation 155
    - IBMSNAP\_SUBS\_COLS 469
    - IBMSNAP\_SUBS\_EVENT 471
    - IBMSNAP\_SUBS\_SET 477
    - IBMSNAP\_SUBS\_STMTS 483
    - SUBS\_MEMBER (membres d'ensembles d'abonnements) 472
  - Tables de contrôle Apply
    - IBMSNAP\_APPLEVEL 455
  - tables de contrôle Capture
    - IBMSNAP\_AUTHTKN 416
    - IBMSNAP\_REG\_EXT 435
      - modification des données 429
  - Tables de contrôle de Capture
    - CAPPARMS (paramètres Capture)
      - modification 136
      - utilisation 133
    - CCD (consistant-change-data) 427
    - IBMSNAP\_CAPENQ 417
    - IBMSNAP\_CAPMON 418
    - IBMSNAP\_CAPPARMS
      - structure 420
    - IBMSNAP\_CAPSCHEMAS 424
    - IBMSNAP\_CAPTRACE 426
    - IBMSNAP\_PARTITIONINFO 431
    - IBMSNAP\_PRUNCNTL 432
    - IBMSNAP\_PRUNE\_LOCK 434
    - IBMSNAP\_PRUNE\_SET 435
    - IBMSNAP\_REG\_SYNCH 444
    - IBMSNAP\_REGISTER 437
    - IBMSNAP\_RESTART 445
    - IBMSNAP\_SEQTABLE 447
    - IBMSNAP\_SIGNAL 447
    - IBMSNAP\_UOW 451
  - tables de contrôle dynamiques 215
  - tables de contrôle statiques 216
  - tables de modification cohérente des données (CCD)
    - ajout de colonnes UOW 87, 487
      - externe
        - réplication multi-niveaux 91
      - internes
        - cibles multiples 89
    - sources de données non relationnelles
      - gestion des tables CCD 67
      - utilisation de tables CCD 45
    - sources de données relationnelles non-DB2
      - utilisation de tables CCD 47
    - sources de réplication 91
    - structure
      - serveur de contrôle Capture 427
    - utilisation
      - historique ou audit 87, 487
      - réplication multi-niveaux 91
      - verrous sur 10
  - tables de modification des données (CD)
    - besoins en mémoire 5
    - élagage 218
    - récapitulatif de contenu 86
    - structure 429
  - tables de signaux de journaux
    - arrêt 203
      - CAPSTOP 206
  - tables définies par l'utilisateur
    - tables cible 96
      - types de tables cible 84
  - tables des points de cohérence
    - structure 489
    - utilisation 85
  - tables du catalogue, enregistrement 45
  - tables maîtres (réplication bidirectionnelle)
    - présentation 94
    - recapture des modifications 56
  - tables partitionnées
    - réplication 34, 36
  - tables partitionnées par intervalle
    - réplication 34, 36
  - tables réplique
    - définition 84
    - définition de cibles en lecture-écriture 94
    - recapture des modifications 56
    - structure 490
  - tables source
    - ajout de colonnes 178
    - création de journaux pour 39
    - gestion 211
    - recupération de données perdues 221
  - tables source distantes 62
  - tâches démarrées par le système 169
  - taille du récepteur
    - actuel 3
  - taille du récepteur actuel 3, 40
  - traduction de données 11
  - traitement d'exécution 78, 114
  - traitement des modifications de type de données 180
  - transactions
    - mémoire utilisée par 1
    - rendement
      - déclencheurs de capture 9
  - transfert
    - données 91
  - transformation de données
    - à l'abonnement 113
    - à l'enregistrement 113
    - changement de nom de colonnes 98, 115
    - création de colonnes calculées 115
  - travaux de journal
    - vérification du statut 250
  - travaux importants de réplication 74
  - travaux par lots
    - exécution 169
    - mémoire 1
  - TSO
    - démarrage du programme Apply 171
    - utilisation de JCL 169
  - type de données BLOB
    - remarques sur la réplication 104
  - type de données CLOB
    - remarques sur la réplication 104
  - type de données DATE
    - réplication 106
  - Type de données DBCLOB
    - remarques sur la réplication 104
  - type de données LOB
    - remarques sur la réplication 104
  - type de données LONG VARCHAR
    - réplication 103
  - type de données LONG VARCHARIC
    - réplication 103
  - type de données NUMBER
    - réplication 106
  - type de données TIMESTAMP
    - réplication 106
  - types de données
    - mappages de colonnes 98
    - réplication
      - objets LOB 104
  - types de données abstraits
    - restrictions pour la réplication 103
  - types de données d'objet BLOB
    - remarques sur la réplication 104
  - types de données distincts 103
  - types de données spéciaux
    - réplication
      - objets LOB 104
  - types UDT
    - restrictions de réplication 103
- ## U
- Unicode
    - tables 493
  - utilitaire ASNPLXFY 175
  - utilitaire asntdiff
    - avec fonctions de compatibilité DB2 232
    - présentation 223
  - utilitaire asntrep
    - guide d'utilisation 232
  - utilitaire asntrepair
    - guide d'utilisation 223
    - présentation 223
  - utilitaire de comparaison des tables
    - exécution en mode parallèle 229
  - utilitaire de différences des tables 223, 290, 294, 302
    - avec fonctions de compatibilité DB2 232
    - type de données DATE
      - réplication 232
    - type de données NUMBER
      - réplication 232
    - type de données TIMESTAMP
      - réplication 232
  - utilitaire de réparation des tables 232, 313
  - utilitaire RUNSTATS
    - réplication SQL 215
  - utilitaires
    - comparaison de tables 229
    - différenciation des tables 290, 294, 302
    - réparation de table 313

## V

- valeurs par défaut
  - des paramètres Apply (Linux, UNIX, Windows, z/OS) 144, 146
  - des paramètres Capture 120
  - des paramètres Capture (Linux, UNIX, Windows, z/OS) 122
  - des paramètres Capture (System
    - i) 120
  - pour des paramètres Apply (System
    - i) 144
- variable d'environnement
  - DB2CODEPAGE 11, 29
- variable d'environnement
  - DB2DBDFT 29
- variable d'environnement DB2INSTANCE
  - réplication 29
- variable d'environnement LANG
  - définition 11
- variable LIBPATH 29
- variables d'environnement
  - DB2CODEPAGE 11, 29
  - DB2DBDFT 29
  - DB2INSTANCE 29
  - LIBPATH 29
  - Programme Capture 29
- variables de registre
  - DB2CODEPAGE 11, 29
  - DB2DBDFT 29
  - DB2INSTANCE 29
- verrous
  - sur des tables CCD 10
- vues
  - enregistrement
    - en tant que sources 67
    - présentation 64
    - procédure 177
  - modification des attributs 178
  - restrictions 64, 67
  - utilisation de l'ID de corrélation 65
- vues DB2
  - enregistrement 67
- vues de modification des données (CD) 64

## W

- warmns startmode 122
- warmsi startmode 122









SC11-7049-00



Spine information:

IBM InfoSphere Data Replication **Version 10.1.3**

**Guide de référence de la réplication SQL**

