

StorageTek Enterprise Library Software

Document de référence de l'interface client XAPI pour serveur ACSLS

E74275-01

Septembre 2015

StorageTek Enterprise Library Software

Document de référence de l'interface client XAPI pour serveur ACSLS

E74275-01

Copyright © 2015, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf stipulation expresse de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, accorder de licence, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer un risque de dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour des applications dangereuses.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée de The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers, sauf mention contraire stipulée dans un contrat entre vous et Oracle. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation, sauf mention contraire stipulée dans un contrat entre vous et Oracle.

Table des matières

Préface	9
Public visé	9
Accessibilité de la documentation	9
Documents connexes	9
Conventions	9
Conventions syntaxiques	10
Lignes de flux	10
Choix requis unique	10
Choix facultatif unique	10
Valeurs par défaut	11
Répétition	11
Mots-clés	11
Variables	12
Autres	12
Facultatif	12
Délimiteurs	12
Plages	12
Plages hexadécimales	12
Plages décimales	12
Plages de VOLSER au format HSC	12
Plages de VOLSER alphanumérique au format ACSLS	14
Listes	14
Espaces	15
1. Introduction	17
Qu'est-ce que l'interface client XAPI pour serveur ACSLS?	17
Qu'est-ce que le composant d'interface XCMD?	17
2. Conventions des commandes XCMD	19
3. Commandes de configuration et d'administration XCMD XAPI	21
DEFINE POOL_Name	21
Description	21

Syntaxe	21
Paramètres	22
Exemple	22
DEFINE POOL_Access	22
Description	22
Syntaxe	23
Paramètres	23
Exemple	24
DELETE POOL_Name	24
Description	24
Syntaxe	24
Paramètres	25
Exemple	25
DELETE POOL_Access	25
Description	25
Syntaxe	25
Paramètres	26
Exemple	26
SET POOL_Name	26
Description	26
Syntaxe	27
Paramètres	27
Exemple	28
QUERY POOL_Name	29
Description	29
Syntaxe	29
Paramètres	29
Exemple	29
QUERY POOL_Access	30
Description	30
Syntaxe	30
Paramètres	30
Exemple	31
4. Commandes d'opérateur du serveur XCMD XAPI	33
LOG	33
Description	33
Syntaxe	33
Paramètres	34

Exemple	35
LIST	35
Description	35
Syntaxe	35
Paramètres	36
Exemple	37
MSGLVL	37
Description	37
Syntaxe	37
Paramètres	38
Exemple	38
TRACE	39
Description	39
Syntaxe	39
Paramètres	39
Exemple	40
5. XCMD prend en charge les demandes d'utilisateur XAPI et les commandes HSC/VTCS	41
Demandes, commandes et paramètres XAPI pris en charge	41
Exécution de commandes vers le serveur ACSLS XAPI avec la commande Route d'un client SMC ou VM	44
6. Variables de contrôle du serveur XAPI ACSLS	45
Configuration des variables de contrôle du serveur XAPI ACSLS	45
Description de variable ACSLS XAPI	46
XAPI_PORT	46
XAPI_WORK_PATH	46
XAPI_LOG_SIZE	47
XAPI_LOG_FILE_NUM	47
XAPI_TRACE_SIZE	47
XAPI_TRACE_FILE_NUM	48
XAPI_STARTUP_FILE	48
XAPI_TAPEPLEX_NAME	48
Index	49

Liste des tableaux

5.1. Demandes XAPI et commandes HSC/MVS correspondantes prises en charge par XCMD	41
-----------------------------------------------------------------------------------------	----

Préface

Cette publication fournit des informations sur la configuration et l'administration d'un serveur Oracle StorageTek ACSLS qui prend en charge XAPI.

Public visé

Ce document s'adresse aux administrateurs de stockage, aux programmeurs système et aux opérateurs chargés de la configuration et de la maintenance d'un serveur ACSLS XAPI.

Accessibilité de la documentation

Pour plus d'informations sur l'engagement d'Oracle pour l'accessibilité à la documentation, visitez le site Web Oracle Accessibility Program, à l'adresse <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Accès aux services de support Oracle

Les clients Oracle qui ont souscrit un contrat de support ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> ou le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> si vous êtes malentendant.

Documents connexes

Pour plus d'informations, reportez-vous aux documents suivants de la documentation d'Oracle StorageTek ELS version 7.3 et de la documentation d'ACSLs version 8.4 :

- *Guide de l'administrateur d'ACSLs 8.4*
- *Guide d'installation d'ACSLs 8.4*
- *Guide de référence rapide ACSLS 8.4*
- *Référence des commandes, des instructions de contrôle et des utilitaires ELS 7.3*
- *Configuration et gestion du SMC 7.3*
- *Guide d'installation, de configuration et d'administration de VM Client*

Conventions

Les conventions de texte suivantes sont utilisées dans ce document :

Convention	Explication
gras	Des caractères gras indiquent des éléments d'interface utilisateur graphique associés à une action ou des termes définis dans le texte ou le glossaire.

Convention	Explication
<i>italique</i>	Les caractères en italique indiquent des titres de livres, la mise en valeur d'un concept ou des variables substituables pour lesquelles vous fournissez des valeurs particulières.
<i>Largeur fixe</i>	Le type largeur fixe indique des commandes au sein d'un paragraphe, des adresses URL, des exemples de code, du texte affiché à l'écran ou du texte que vous saisissez.

Conventions syntaxiques

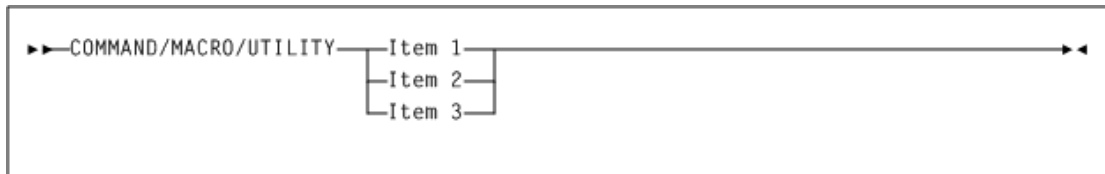
Remarque:

Les conventions syntaxiques suivantes s'appliquent **exclusivement** à la syntaxe XAPI et XCMD. Reportez-vous au *Guide de l'administrateur d'ACSL* pour obtenir des informations sur les conventions syntaxiques qui s'appliquent aux commandes et utilitaires d'ACSL.

Les conventions de diagrammes de flux syntaxiques incluent ce qui suit :

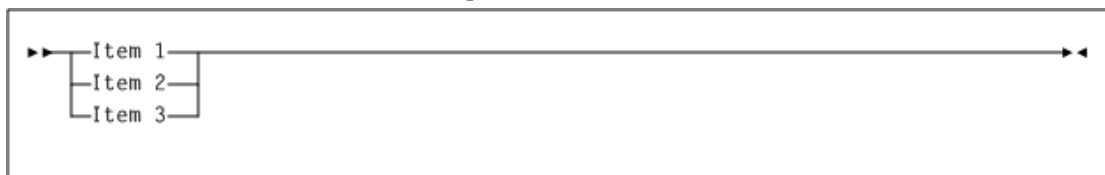
Lignes de flux

Les diagrammes syntaxiques sont constitués d'une ligne de base horizontale, de lignes de branches horizontales et verticales et du texte d'une commande, d'une instruction de contrôle, d'une macro ou d'un utilitaire. Les diagrammes se lisent de gauche à droite et de haut en bas. Les flèches indiquent le flux et la direction. Par exemple :



Choix requis unique

Les lignes de branches indiquent qu'un seul choix doit être effectué. Si l'un des éléments parmi lesquels effectuer un choix est positionné sur la ligne de base d'un diagramme, un élément doit être sélectionné. Par exemple :



Choix facultatif unique

Si le premier élément est positionné sur la ligne en dessous de la ligne de base, un élément peut être sélectionné si cela est souhaité. Par exemple :



Valeurs par défaut

Les valeurs et les paramètres par défaut apparaissent au-dessus de la ligne de base. Par exemple :

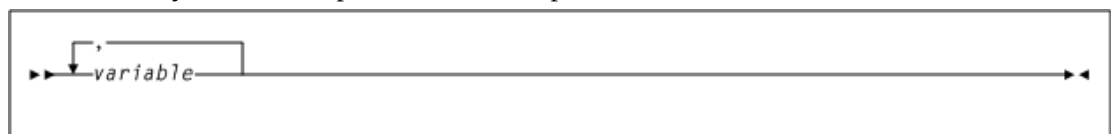


Certains paramètres de mots-clés offrent un choix de valeurs dans une pile. Lorsque la pile contient une valeur par défaut, les choix de mot-clé et de valeur sont placés en dessous de la ligne de base pour indiquer qu'ils sont facultatifs et la valeur par défaut apparaît au-dessus de la ligne de mot-clé. Par exemple :



Répétition

Un symbole de répétition indique qu'il est possible d'effectuer plusieurs choix ou d'effectuer un seul choix plusieurs fois. L'exemple suivant indique qu'une virgule est nécessaire pour délimiter les symboles de répétition. Par exemple :



Mots-clés

Tous les mots-clés de commandes apparaissent intégralement en majuscules ou à la fois en majuscules et en minuscules. Lorsque les commandes ne font pas la distinction entre majuscules et minuscules, utiliser à la fois des majuscules et des minuscules implique que les minuscules peuvent être omises pour former une abréviation.

Variables

Le texte en italique indique une variable.

Autres

Une barre oblique (|) sépare les valeurs de paramètres alternatives.

Facultatif

Les crochets [] indiquent qu'un paramètre de commande est facultatif.

Délimiteurs

Si une virgule (,), un point virgule (;) ou autre délimiteur apparaît avec un élément du diagramme syntaxique, il doit être entré comme faisant partie de l'instruction.

Plages

Une plage inclusive est indiquée par une paire d'éléments ayant la même longueur et le même type de données, liés par un tiret. Le premier élément doit être strictement inférieur au second.

Plages hexadécimales

Une plage hexadécimale consiste en une paire de nombres hexadécimaux (par exemple, 0A2-0AD ou 000-0FC).

Plages décimales

Une plage décimale consiste en une paire de nombres décimaux (par exemple, 1-9 ou 010-094). Les zéros à gauche ne sont pas nécessaires. La portion décimale est une plage incrémentielle. Les positions des caractères de la portion incrémentielle des deux éléments de plages doivent correspondre et les caractères non incrémentiels du premier élément doivent être identiques à ceux du second élément.

Plages de VOLSER au format HSC

Une plage de numéros de série de volumes (VOLSER) numériques au format HSC consiste en une paire d'éléments VOLSER contenant une portion numérique décimale de 1 à 6 chiffres (par exemple, ABC012-ABC025 ou X123CB-X277CB). La portion décimale est une plage incrémentielle. Les restrictions supplémentaires suivantes s'appliquent :

- Les positions des caractères de la portion incrémentielle des deux éléments de plages doivent correspondre.

- Les caractères non incrémentiels du premier élément doivent être identiques à ceux du second élément.
- Il n'est pas possible d'incrémenter deux portions d'un élément de plage. Si 111AAA est le premier élément, vous ne pouvez pas spécifier 112AAB pour le second élément.
- Si une plage VOLSER contient plusieurs portions décimales, n'importe quelle portion est valide en tant que plage incrémentielle. Par exemple :
 - A00B00 - La plage la plus grande pouvant être spécifiée va de A00B00 à A99B99.
 - A0B0CC - La plage la plus grande pouvant être spécifiée va de A0B0CC à A9B9CC.
 - 000XXX - La plage la plus grande pouvant être spécifiée va de 000XXX à 999XXX.

Une plage VOLSER alphabétique au format HSC consiste en une paire d'éléments VOLSER contenant une portion incrémentielle de 1 à 6 caractères (par exemple, 000AAA-000ZZZ ou 9AAA55-9ZZZ55). Cette portion est une plage incrémentielle. Les restrictions supplémentaires suivantes s'appliquent :

- Les positions des caractères de la portion incrémentielle des deux éléments de plages doivent correspondre.
- Les caractères non incrémentiels du premier élément doivent être identiques à ceux du second élément.
- Il n'est pas possible d'incrémenter deux portions d'un élément de plage. Si 111AAA est le premier élément, vous ne pouvez pas spécifier 112AAB pour le second élément.
- La portion alphabétique de la plage VOLSER s'étend du caractère A au caractère Z. Pour incrémenter des séquences à plusieurs caractères, chaque élément s'incrémente à Z. Par exemple, ACZ fait partie de la plage AAA-AMM. En voici quelques exemples :
 - A00A0-A99A0
 incrémente les plages VOLSER A00A0 à A09A0, puis A10A0 à A99A0.
 - 9AA9A-9ZZ9A
 incrémente les plages VOLSER 9AA9A à 9AZ9A, puis 9BA9A à 9ZZ9A.
 - 111AAA-111ZZZ
 incrémente les plages VOLSER 111AAA à 111AAZ, puis 111ABA à 111ZZZ
 - 999AM8-999CM8
 incrémente les plages VOLSER 999AM8 à 999AZ8, puis 999BA8 à 999CM8
 - A3BZZ9-A3CDE9
 incrémente les plages VOLSER A3BZZ9 à A3CAA9, puis A3CAB9 à A3CDE9
 - AAAAAA-AAACCC
 incrémente les plages VOLSER AAAAAA à AAAAAZ, puis AAAABA à AAACCC
 - CCCNNN-DDDNNN

incrémente les plages VOLSER CCCNNN à CCCNNZ, puis CCCNOA à DDDNNN. Il s'agit d'une plage très grande.

Le nombre de volumes dans une plage VOLSER alphabétique dépend du nombre d'éléments dans la portion incrémentielle de la plage VOLSER. Pour une plage A à Z à chaque position de caractère, le nombre de volumes peut être calculé par 26 à la puissance du nombre de positions incrémentées.

- A-Z équivaut à 26^1 ou 26 volumes.
- AA-ZZ équivaut à 26^2 ou 676 volumes.
- AAA-ZZZ équivaut à 26^3 ou 17 576 volumes.
- AAAA-ZZZZ équivaut à 26^4 ou 456 976 volumes.
- AAAAA-ZZZZZ équivaut à 26^5 ou 11 881 376 volumes.
- AAAAAA-ZZZZZZ équivaut à 26^6 ou 308 915 776 volumes.

Plages de VOLSER alphanumérique au format ACSLS

Une plage de VOLSER alphanumérique au format ACSLS n'a pas à être conforme au modèle de plages de volumes au format HSC spécifié ci-dessus. Les restrictions supplémentaires suivantes s'appliquent :

- Les deux VOLSER limitant la plage doivent contenir le même nombre de caractères.
- Le premier élément VOLSER de la plage doit être strictement inférieur au second élément VOLSER.
- Si les VOLSER spécifiés comportent moins de 6 caractères, les VOLSER sont évalués comme s'ils comportaient des espaces à gauche pour compléter les 6 caractères.
- Seuls les VOLSER qui sont strictement supérieurs ou égaux au premier VOLSER et strictement inférieurs ou égaux au deuxième VOLSER appartiennent à la plage lorsque celle-ci est évaluée par ordre de classement ACSII. Par exemple, la plage de VOLSER 123456-234567 contient le VOLSER 12345A.

Remarque:

- Une plage de VOLSER au format HSC valide est systématiquement valide au regard des règles de VOLSER au format ACSLS. Cependant, une plage de VOLSER au format ACSLS n'est pas forcément valide au regard des règles de VOLSER au format HSC.
 - La plupart des commandes XCMD qui reconnaissent les plages de VOLSER évaluent ses dernières conformément aux règles de VOLSER HSC. La commande XCMD SET POOL_Name est la seule commande qui accepte les deux formats de plage de VOLSER.
-

Listes

Une liste consiste en un ou plusieurs éléments. Si plusieurs éléments sont spécifiés, ils doivent être séparés par une virgule ou un espace et la liste entière doit être incluse entre parenthèses.

Espaces

Les valeurs et les paramètres de mots-clés peuvent être séparés par un nombre quelconque d'espaces.

Chapitre 1. Introduction

Ce chapitre présente l'interface client XAPI avec ACSLS et décrit l'interface XCMD qui permet de saisir des commandes ciblant un serveur ACSLS avec XAPI.

Qu'est-ce que l'interface client XAPI pour serveur ACSLS?

L'interface XML API (XAPI) est l'API StorageTek d'Oracle qui permet aux clients et aux serveurs StorageTek de communiquer à l'aide d'un protocole commun sur TCP/IP.

Avec l'introduction de XAPI, les clients qui auparavant devaient utiliser un serveur basé sur MVS (StorageTek Host software Component d'Oracle) pour le traitement de bandes physiques peuvent désormais utiliser ACSLS (avec la prise en charge XAPI activée) comme suit :

- Un client SMC sur MVS peut maintenant faire des demandes de bandes réelles à partir d'un serveur ACSLS avec support XAPI activé (sans avoir recours à MVS/CSC).

Pour plus d'informations, reportez-vous à la publication ELS *Configuration et gestion du SMC*.

- Un système VM Client peut maintenant demander des services de bandes réelles à partir d'un serveur ACSLS avec support XAPI activé.

Pour plus d'informations, reportez-vous à la publication ELS *Guide d'installation, de configuration et d'administration de VM Client*.

Le service XAPI est pris en charge par ACSLS 8.4 et versions ultérieures. Reportez-vous au *Guide d'installation StorageTek ACSLS 8.4* pour obtenir plus d'informations sur l'installation et la configuration du service ACSLS XAPI.

Qu'est-ce que le composant d'interface XCMD?

Le composant ACSLS cmd_proc est une interface de ligne de commande d'ACSLs qui permet aux clients de gérer un système ACSLS et les bibliothèques contrôlées par ACSLS.

Reportez-vous au chapitre Présentation du *Guide de l'administrateur de StorageTek Automated Cartridge System Library Software* pour plus d'informations relatives à l'utilisation de cmd_proc. La commande XCMD est une extension d'ACSLs cmd_proc qui permet aux utilisateurs du composant de serveur ACSLS XAPI de saisir un nombre limité

de demandes d'utilisateur et d'administrateur du serveur XAPI à partir du composant ACSLS cmd_proc existant lorsque le composant de serveur ACSLS XAPI est installé et actif.

Reportez-vous à la section [Chapitre 3, Commandes de configuration et d'administration XCMD XAPI](#) pour plus d'informations sur les commandes de serveur XAPI que vous pouvez entrer à partir d'ACSLs cmd_proc avec la commande *XCMD*.

Reportez-vous à la section [Chapitre 4, Commandes d'opérateur du serveur XCMD XAPI](#) pour plus d'informations sur les commandes de serveur XAPI que vous pouvez entrer à partir d'ACSLs cmd_proc avec la commande *XCMD*.

En outre, vous pouvez utiliser la commande *Route* du SMC/MVS ou la commande *Route* du client VM pour exécuter les commandes d'opérateur XAPI, les commandes de configuration et d'administration XCMD et les commandes HSC/VTCS XAPI à partir d'un hôte MVS ou VM. Cependant, certaines restrictions de paramètre s'appliquent.

Reportez-vous à la section [Chapitre 5, XCMD prend en charge les demandes d'utilisateur XAPI et les commandes HSC/VTCS](#) pour obtenir des informations sur les commandes HSC/VTCS et d'autres demandes XAPI prises en charge, ainsi que sur les restrictions de paramètre.

Chapitre 2. Conventions des commandes XCMD

La majorité des interactions client/serveur entre clients SMC/MVS et VM et un serveur ACSLS avec XAPI sont transparentes pour l'utilisateur final. Les demandes relatives aux informations de volumes, aux montages et démontages sont générées automatiquement par les clients SMC/MVS et VM, et sont traitées sans l'intervention d'un opérateur.

En plus de ces interactions automatiques, le serveur ACSLS avec XAPI fournit des commandes d'administrateur, de configuration et d'opérateur supplémentaires qui vous permettent de gérer le composant XAPI.

- Reportez-vous à la section [Chapitre 3, Commandes de configuration et d'administration XCMD XAPI](#) pour obtenir des informations sur les commandes de configuration et d'administration du serveur XCMD XAPI.
- Reportez-vous à la section [Chapitre 4, Commandes d'opérateur du serveur XCMD XAPI](#) pour obtenir des informations sur les commandes d'opérateur du serveur XCMD XAPI.

Ces commandes n'appartiennent pas à l'installation de base d'ACSLs, mais sont disponibles lorsque le composant ACSLS XAPI est installé en option. Reportez-vous au *Guide d'installation de StorageTek Automated Cartridge System Library Software ACSLS* pour obtenir des informations sur l'installation d'ACSLs et du composant ACSLS XAPI.

Pour exécuter ces commandes, utiliser l'utilitaire de commande XCMS exécuté à partir d'ACSLs cmd_proc. La syntaxe générale de cette commande est la suivante :

XCMD commande

Où *commande* est l'une des commandes du composant ACSLS XAPI décrites dans cette publication.

Remarque:

Vous pouvez également utiliser l'utilitaire de commande XCMD pour exécuter toute demande d'utilisateur XAPI prise en charge par le serveur ACSLS XAPI. Pour plus d'informations, reportez-vous au [Chapitre 5, XCMD prend en charge les demandes d'utilisateur XAPI et les commandes HSC/VTCS](#).

Les conventions suivantes s'appliquent dès que vous exécutez une commande XCMD à partir d'ACSLs cmd-proc :

- XCMD n'utilise pas l'invite de commande ACSLS.

-
- Lorsque vous saisissez des paramètres de mot-clé avec une valeur associée, la valeur doit être précédée d'un seul caractère vide.
 - Les commandes XCMD ne sont pas sensibles à la casse. Le composant ACSLS XAPI convertit automatiquement toutes les commandes en lettres majuscules. Par exemple, les noms de sous-pool "Testpool1", "testpool1", "TestPool1" et "TESTPOOL1" sont tous convertis en "TESTPOOL1".

Remarque:

Les commandes et paramètres ACSLS cmd_proc normaux sont généralement saisis en lettres minuscules. Les lettres minuscules ne sont pas automatiquement converties en lettres majuscules, exception faite des vol_id (VOLSER).

- Les paramètres XCMD ne doivent pas être précédés d'un signe égal (=).
- Les valeurs XCMD ne doivent pas être comprises entre parenthèses. Lorsque plusieurs valeurs sont prises en charge, vous pouvez les séparer par des virgules. Par exemple, pour interroger plusieurs volumes, saisissez XCMD QUERY VOLUME volser1,volser2 et non XCMD QUERY VOLUME(volser1,volser2).

Chapitre 3. Commandes de configuration et d'administration XCMD XAPI

Ce chapitre décrit les commandes de configuration et d'administration XAPI prises en charge par XCMD. Ces commandes activent les fonctionnalités de pool de volumes compatible ELS dans ACSLS version 8.4. Les commandes prises en charge sont les suivantes :

- [DEFINE POOL_Name](#)
- [DEFINE POOL_Access](#)
- [DELETE POOL_Name](#)
- [DELETE POOL_Access](#)
- [SET POOL_Name](#)
- [QUERY POOL_Name](#)
- [QUERY POOL_Access](#)

Exécutez ces commandes à partir d'ACSLs user_proc avec la commande XCMD. Pour plus d'informations, voir [Chapitre 2, Conventions des commandes XCMD](#) .

Reportez-vous à la section "[Conventions syntaxiques](#)" quant aux conventions de diagrammes de flux syntaxique inclus dans les sections de commande suivantes.

DEFINE POOL_Name

La section suivante décrit la commande *DEFINE POOL_Name*.

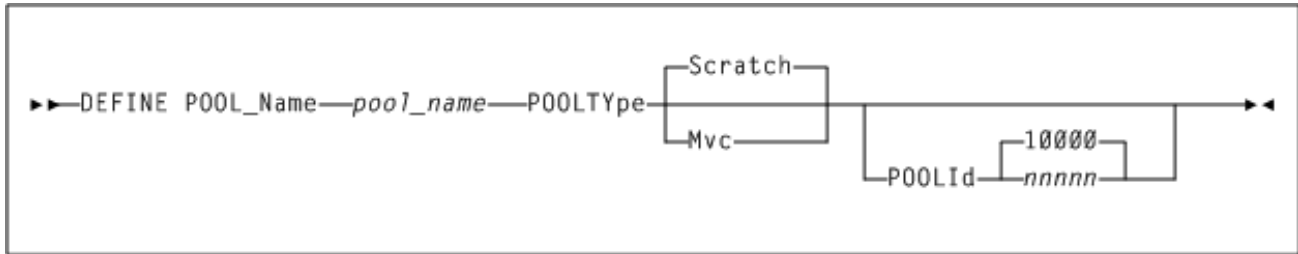
Description

La commande *DEFINE POOL_NAME* associe un type et un nom de pool à un ID pool ACSLS.

Syntaxe

La figure suivante illustre la syntaxe de la commande *DEFINE POOL_Name* :

Figure 3.1. Syntaxe de la commande DEFINE POOL_Name



Paramètres

Comme illustré dans [Figure 3.1, « Syntaxe de la commande DEFINE POOL_Name »](#), la commande `DEFINE POOL_Name` comprend les paramètres suivants :

pool_name

Nom de pool ne dépassant pas 13 caractères. Le nom de pool peut inclure n'importe quel caractère, sauf le symbole pour cent ("%"), le trait de soulignement ("_") et l'astérisque ("*").

POOLType

Type de pool, soit *Scratch* (par défaut) ou *Mvc*.

POOLId (nnnnn)

Indique facultativement l'ID pool (*nnnnn*).

- Si vous choisissez de spécifier ce paramètre, vous devez indiquer un ID pool qui n'existe pas encore.
- Si vous ne spécifiez pas ce paramètre, le système attribuera automatiquement un ID pool en tant que première valeur non utilisée à compter de 10000.

Exemple

Dans l'exemple suivant, un utilisateur exécute la commande `DEFINE POOL_Name` pour définir le pool de travail `SCRPOOL1` :

```
XCMD DEFINE POOL_NAME SCRPOOL1 POOLTYPE SCRATCH
```

DEFINE POOL_Access

La section suivante décrit la commande `DEFINE POOL_Access`.

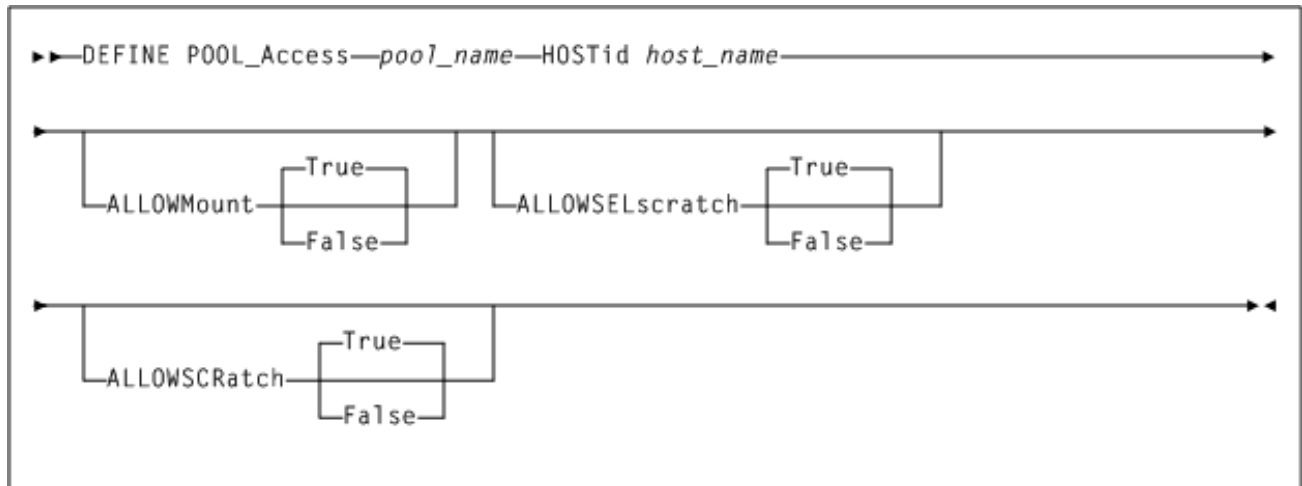
Description

La commande `DEFINE POOL_Access` définit l'accès accordé à un *pool_name* par un hôte spécifié. Si aucun accès de pool n'est défini pour un *pool_name* spécifique, tous les accès sont autorisés.

Syntaxe

La figure suivante illustre la syntaxe de la commande `DEFINE POOL_Access` :

Figure 3.2. Syntaxe de la commande `DEFINE POOL_Access`



Paramètres

Comme illustré dans [Figure 3.2, « Syntaxe de la commande DEFINE POOL_Access »](#), la commande `DEFINE POOL_Access` comprend les paramètres suivants :

pool_name

Nom de pool ne dépassant pas 13 caractères. Le nom de pool peut inclure n'importe quel caractère, sauf le symbole pour cent ("%"), le trait de soulignement ("_") et l'astérisque ("*"). Le *pool_name* spécifié doit avoir été défini précédemment au moyen de la commande `DEFINE POOL_Name` avec un type de pool `SCRATCH` (l'accès n'est pas défini pour les pools `MVC`). Pour plus d'informations, reportez-vous au "[DEFINE POOL_Name](#)".

HOSTid host_name

Spécifie un nom d'hôte ne dépassant pas 31 caractères. Ce paramètre est obligatoire.

Le *HOSTid* spécifié correspond à la portion de nom d'hôte du nom de domaine qualifié complet d'Internet. Par exemple, si votre nom de domaine qualifié complet est *myhost.plus.domain.name*, le *HOSTid* spécifié est *myhost*.

ALLOWMount

facultativement, indique l'accès de montage. Vous pouvez indiquer *True* ou *False*. Ce paramètre n'est valide que pour un pool défini comme `POOLTYPE MVC`.

ALLOWSELscratch

facultativement, indique l'accès des commandes XAPI `query_scr_mnt_info`, `sel_scratch` et `mount` pour un volume de travail. Vous pouvez indiquer *True* ou *False*. Si vous ne spécifiez pas ce paramètre, il est défini sur *True* par défaut.

Ce paramètre n'est valide que pour un pool défini comme *POOLTYPE SCRATCH*. Si vous spécifiez ce paramètre pour un pool défini comme *MVC*, le paramètre est automatiquement défini par défaut sur *False*.

ALLOWSCRatch

facultativement, indique l'accès de travail. Vous pouvez indiquer *True* ou *False*. Si vous ne spécifiez pas ce paramètre, il est défini sur *True* par défaut.

Ce paramètre n'est valide que pour un pool défini comme *POOLTYPE SCRATCH*. Si vous spécifiez ce paramètre pour un pool défini comme *MVC*, le paramètre est automatiquement défini par défaut sur *False*.

Exemple

Dans l'exemple suivant, un utilisateur exécute la commande *DEFINE POOL_Access* pour définir l'accès en fonction du nom de pool de travail *SCRPOOL1*, où l'hôte est *PRODMVS* et *ALLOWMOUNT* et *ALLOWSCRATCH* sont définis sur *True*:

```
XCMD DEFINE POOL_ACCESS SCRPOOL1 HOSTID PRODMVS ALLOWMOUNT TRUE
ALLOWSCRATCH TRUE
```

DELETE POOL_Name

La section suivante décrit la commande *DELETE POOL_Name*.

Description

La commande *DELETE POOL_Name* supprime un pool. Cette commande est similaire à la commande *delete pool* d'ACSLs, avec les exceptions suivantes :

- L'entrée spécifie un *nom de pool* au lieu d'un *ID pool*.
- L'utilisateur a la possibilité de réattribuer les volumes attribués au pool 0.

Reportez-vous au *Guide de l'utilisateur d'Automated Cartridge System Library Software (ACSLs)* de *StorageTek* pour plus d'informations sur la commande *delete pool* d'ACSLs.

Syntaxe

La figure suivante illustre la syntaxe de la commande *DELETE POOL_Name* :

Figure 3.3. Syntaxe de la commande DELETE POOL_Name



Paramètres

Comme illustré dans [Figure 3.3, « Syntaxe de la commande DELETE POOL_Name »](#), la commande `DELETE POOL_Name` comprend les paramètres suivants :

pool_name

Nom de pool ne dépassant pas 13 caractères. Le nom de pool peut inclure n'importe quel caractère, sauf le symbole pour cent ("%"), le trait de soulignement ("_") et l'astérisque ("*"). Le *pool_name* spécifié doit avoir été défini précédemment au moyen de la commande `DEFINE POOL_Name`. Pour plus d'informations, reportez-vous au "[DEFINE POOL_Name](#)".

`RESETvols`

Indique que les volumes actuellement attribués au pool en cours de suppression doivent être réattribués au pool 0. La commande `delete pool-name` échoue si vous ne spécifiez pas ce paramètre alors que les volumes existent dans le pool.

Exemple

Dans l'exemple suivant, un utilisateur exécute la commande `DELETE POOL_Name` pour supprimer le pool `SCRPOOL1` et réattribuer ses volumes au pool 0 (le pool par défaut) :

```
XCMD DELETE POOL_NAME SCRPOOL_TEST1 RESETVOLS
```

DELETE POOL_Access

La section suivante décrit la commande `DELETE POOL_Access`.

Description

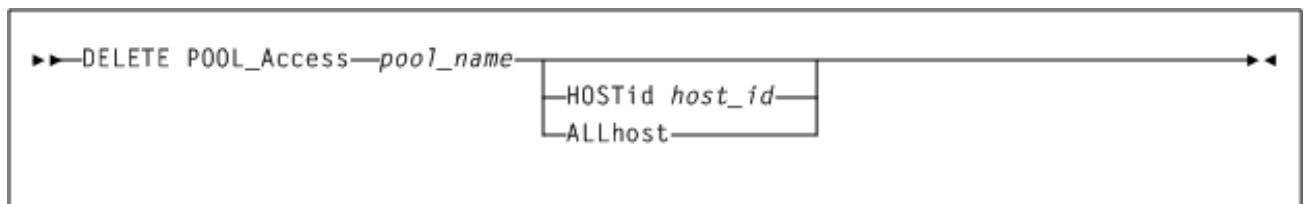
La commande `DELETE POOL_Access` supprime les règles d'accès au pool associées à un seul nom d'hôte (*host_id*) ou supprime toutes les règles d'accès au pool pour tous les hôtes.

- Si un nom d'hôte (*host_id*) est spécifié, seules les règles d'accès de l'hôte spécifié sont supprimées du pool spécifié.
- Si `ALLhost` est spécifié, toutes les règles d'accès au pool sont supprimées.

Syntaxe

La figure suivante illustre la syntaxe de la commande `DELETE POOL_Access` :

Figure 3.4. Syntaxe de la commande DELETE POOL_Access



Paramètres

Comme illustré dans [Figure 3.4, « Syntaxe de la commande DELETE POOL_Access »](#), la commande `DELETE POOL_Access` comprend les paramètres suivants :

pool_name

Nom de pool ne dépassant pas 13 caractères. Le nom de pool peut inclure n'importe quel caractère, sauf le symbole pour cent ("%"), le trait de soulignement ("_") et l'astérisque ("*"). Le *pool_name* spécifié doit avoir été défini précédemment au moyen de la commande `DEFINE POOL_Name`. Pour plus d'informations, reportez-vous au "[DEFINE POOL_Name](#)".

HOSTid *host_name*

facultativement, spécifie un nom d'hôte ne dépassant pas 31 caractères et indique que les restrictions d'accès pour cet ID hôte doivent être supprimées (les règles d'accès de tout autre ID hôte restent inchangées). Ce paramètre et le paramètre *ALLhost* s'excluent mutuellement

ALLhost

facultativement, indique que toutes les restrictions d'accès du *pool_name* spécifique sont supprimées pour tous les ID hôte. Ce paramètre et le paramètre *HOSTid* s'excluent mutuellement.

Exemple

Dans l'exemple suivant, un utilisateur exécute la commande `DELETE POOL_Name` pour supprimer l'accès au pool de l'hôte *PRODMVS* à *SCRPOOL1* :

```
XCMD DELETE POOL_ACCESS SCRPOOL1 HOSTID PRODMVS
```

SET POOL_Name

La section suivante décrit la commande `SET POOL_Name`.

Description

La commande `SET POOL_Name` attribue des volumes à un nom de pool ou définit la limite inférieure du contrôle du débit pour un nom de pool.

Remarque:

Si la syntaxe des commandes est valide, les plages de volumes spécifiées sont ajoutées à la base de données même si certains volumes appartenant à la plage ne peuvent pas être ajoutés au pool. Les volumes ne peuvent pas être ajoutés au pool si l'une des conditions suivantes se vérifie :

- Le volume est *VERROUILLÉ*.
 - Le volume se trouve déjà dans un autre pool, à moins que *FORCE* ne soit spécifié.
 - Le volume se trouve dans une bibliothèque logique.
 - Le volume a un propriétaire ACSLS, et le contrôle d'accès est activé.
-

données, même si certains volumes de la plage ne peuvent pas être attribués au pool. Les plages de volumes spécifiées ne doivent ni chevaucher d'autres plages de volumes, ni se chevaucher entre elles.

Les plages de volumes sont évaluées selon les plages VOLSER au format ACSLS :

- Les deux VOLSER limitant la plage doivent contenir le même nombre de caractères.
- Le premier élément VOLSER de la plage doit être strictement inférieur au second élément VOLSER.
- Si les VOLSER spécifiés comportent moins de 6 caractères, les VOLSER sont évalués comme s'ils comportaient des espaces à gauche pour compléter les 6 caractères.
- Seuls les VOLSER qui sont strictement supérieurs ou égaux au premier VOLSER et strictement inférieurs ou égaux au deuxième VOLSER appartiennent à la plage lorsque celle-ci est évaluée par ordre de classement ACSII. Par exemple, la plage de VOLSER 123456-234567 contient le VOLSER 12345A.

Si les plages spécifiées sont valides, elles sont ajoutées à la base de données *pool_name*, même si certains volumes de la plage ne peuvent pas être attribués au pool. Les plages de volumes spécifiées ne doivent ni chevaucher d'autres plages de volumes, ni se chevaucher entre elles.

ADD

Indique que les plages de volumes spécifiées vont être ajoutées aux plages existantes du pool. Il s'agit de la valeur par défaut. Si cette valeur est spécifiée, *VOLUMES* doit aussi être spécifié. *ADD* et *REPLACE* s'excluent mutuellement.

REPLACE

Indique que les plages de volumes spécifiées vont remplacer les plages de volumes actuellement spécifiées, et que les volumes des plages précédentes seront réattribués au pool 0. Si spécifié, *VOLUMES* (ou *VOLSER*) doit aussi être spécifié. *REPLACE* et *ADD* s'excluent mutuellement.

FORCE

Facultativement, indique que les volumes déjà attribués à d'autres pools doivent être réattribués au nouveau pool. Si ce paramètre n'est pas spécifié, tout volume actuellement attribué à un autre pool n'est pas mis à jour. Si spécifié, *VOLUMES* (ou *VOLSER*) doit aussi être spécifié.

APPLY

Facultativement, indique que les plages de volumes précédemment définies doivent être appliquées aux volumes dans la base de données. Ce paramètre doit être utilisé lorsque certains volumes n'ont pas été mis à jour du fait des restrictions répertoriées ci-dessus. Si spécifié, *VOLUMES* (ou *VOLSER*) doit aussi être spécifié.

Exemple

Dans l'exemple suivant, la commande ajoute les volumes *DRL001-DRL999* à *SCRPOOL1* :

```
XCMD SET POOL_NAME SCRPOOL_TEST1 VOLUMES DRL001-DRL999
```

QUERY POOL_Name

La section suivante décrit la commande *QUERY POOL_Name*.

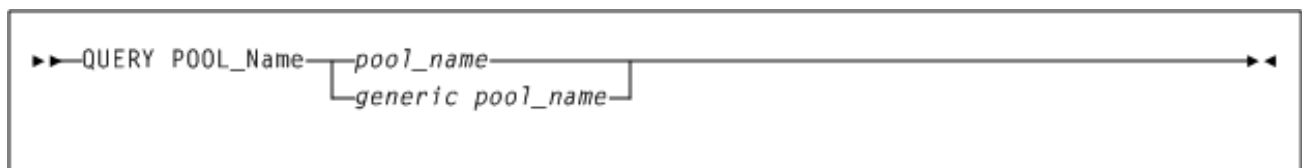
Description

La commande *QUERY POOL_Name* affiche tous les pools, les pools dont le nom commence par une chaîne de caractères précises ou un seul pool.

Syntaxe

La figure suivante illustre la syntaxe de la commande *QUERY POOL_Name* :

Figure 3.6. Syntaxe de la commande QUERY POOL_Name



Paramètres

Comme illustré dans [Figure 3.6, « Syntaxe de la commande QUERY POOL_Name »](#), la commande *QUERY POOL_Name* comprend les paramètres suivants :

pool_name

Soit le nom d'un pool existant ne dépassant pas 13 caractères ou un nom générique, suivi d'un astérisque ("*"), Ce paramètre est obligatoire.

Le format *pool_name* détermine l'affichage de sortie comme suit :

- Lorsque *pool_name* n'est pas générique, les informations de pool affichées comprennent les VOLSER définis et les plages de VOLSER de l'unique pool correspondant.
- Lorsque *pool_name* est générique (suivi de "*"), les informations de pool affichées ne comprennent pas les VOLSER définis et les plages de VOLSER des pools correspondants.

Exemple

Dans l'exemple suivant, un utilisateur exécute la commande *QUERY POOL_Name* pour afficher les informations concernant pool1, y compris les plages de volumes associées.

```
XCMD QUERY POOL_NAME POOL1
```

Dans l'exemple suivant, un utilisateur exécute la commande *QUERY POOL_Name* pour afficher les informations de tous les pools nommés (les plages de volumes ne sont pas incluses).

```
XCMD QUERY POOL_NAME *
```

Dans l'exemple suivant, un utilisateur exécute la commande *QUERY POOL_Name* pour afficher les informations de tous les pools dont le nom commence par "A" (les plages de volumes ne sont pas incluses).

```
XCMD QUERY POOL_NAME A*
```

QUERY POOL_Access

La section suivante décrit la commande *QUERY POOL_Access*.

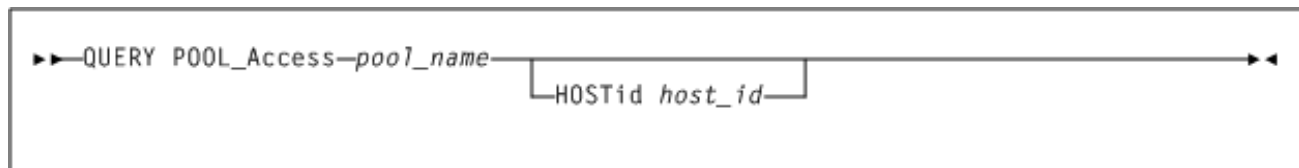
Description

La commande *QUERY POOL_Access* affiche les règles d'accès au pool d'un seul hôte ou de tous les hôtes associés à un nom de pool.

Syntaxe

La figure suivante illustre la syntaxe de la commande *QUERY POOL_Access* :

Figure 3.7. Syntaxe de la commande QUERY POOL_Access



Paramètres

Comme illustré dans [Figure 3.7, « Syntaxe de la commande QUERY POOL_Access »](#), la commande *QUERY POOL_Access* comprend les paramètres suivants :

pool_name

Nom de pool ne dépassant pas 13 caractères. Le nom de pool peut inclure n'importe quel caractère, sauf le symbole pour cent ("%"), le trait de soulignement ("_") et l'astérisque (*). Le *pool_name* spécifié doit avoir été défini précédemment au moyen de la commande *DEFINE POOL_Name*. Pour plus d'informations, reportez-vous au "[DEFINE POOL_Name](#)".

HOSTid host_id

facultativement, indique un *host_name* comportant de 1 à 31 caractères. Le *host_id* spécifié doit avoir été défini précédemment pour *pool_name* au moyen de la commande *DEFINE_POOL_Access*. Pour plus d'informations, voir "[DEFINE POOL_Access](#)".

Si vous ne renseignez pas ce paramètre, les règles d'accès de tous les hôtes associés au pool s'affichent.

Exemple

Dans l'exemple suivant, un utilisateur exécute la commande *QUERY POOL_Access* pour interroger les règles d'accès de tous les hôtes associés au pool_name *SCRPOOL1* :

```
XCMD QUERY POOL_ACCESS SCRPOOL1
```

Chapitre 4. Commandes d'opérateur du serveur XCMD XAPI

Ce chapitre décrit les commandes d'opérateur XAPI prises en charge par XCMD. Ces commandes permettent à un opérateur ou à un administrateur de contrôler le composant ACSLS XAPI. Les commandes prises en charge sont les suivantes :

- [LOG](#)
- [LIST](#)
- [MSGLVL](#)
- [TRACE](#)

Exécutez ces commandes à partir d'ACSLs user_proc avec la commande *XCMD*. Pour plus d'informations, reportez-vous à la section [Chapitre 2, Conventions des commandes XCMD](#) .

Reportez-vous à la section " [Conventions syntaxiques](#) " quant aux conventions de diagrammes de flux syntaxique inclus dans les sections de commande suivantes.

LOG

La section suivante décrit la commande *LOG*.

Description

La commande *LOG* modifie ou répertorie les paramètres actuels du journal du serveur XAPI.

Remarque:

Le nom et l'emplacement du fichier journal sont définis par les variables environnementales ACSLS. Par défaut, le fichier est nommé *vlog.file* et se trouve dans le répertoire *XAPI_WORK_PATH*.

Syntaxe

La figure suivante illustre la syntaxe de la commande *LOG* :

Figure 4.1. Syntaxe de la commande LOG



Paramètres

Comme illustré dans la [Figure 4.1, « Syntaxe de la commande LOG »](#), la commande *LOG* inclut les paramètres suivants :

LIst

facultativement, indique que le paramètre du journal actuel est affiché. Si aucun paramètre de commande n'est spécifié, la valeur par défaut est *LIst*.

log_setting

facultativement, indique le paramètre de journal souhaité sous la forme d'une série de "0" et de "1".

- 0 indique que le paramètre positionnel de journal souhaité doit correspondre à *OFF* ou être désactivé.
- 1 indique que le paramètre positionnel de journal souhaité doit correspondre à *ON* ou être activé.

La valeur spécifiée remplace complètement et n'est pas fusionnée avec le paramètre de journal actuel. Les paramètres positionnels de journal sont les suivants :

- 1 - Consigner les messages d'erreur dans le journal du composant ACSLS XAPI.
- 01 - Consigner les messages dans le fichier journal
- 001 - Consigner les erreurs de transaction d'entrée dans le fichier journal
- 0001 - Consigner tous les paquets *recv()* dans le fichier journal.
- 00001 - Consigner tous les paquets *send()* dans le fichier journal.
- 000001 - Consigner les commandes locales et les réponses dans le fichier journal.

Les messages *LOG* de XAPI seront enregistrés dans *\$ACS_HOME/log/xapi/vlog.file*.

Remarque:

La commande *LOG* vous permet de définir et d'afficher plus de positions de "0" et de "1" (16) que celles définies actuellement dans les paramètres *LOG* ci-dessus ; cela permet de futures expansions, et tout "0" ou "1" superflues du paramètre *LOG* sera ignoré.

OFF

facultativement, indique que la journalisation doit être désactivée. C'est équivalent à *LOG 0*.

Exemple

Dans l'exemple suivant, un utilisateur exécute la commande *LOG* pour configurer le serveur XAPI de manière à consigner toutes les erreurs et tous les paquets *recv()* XML dans le fichier *journal* :

```
XCMD LOG 1011
```

LIST

La section suivante décrit la commande *LIST*.

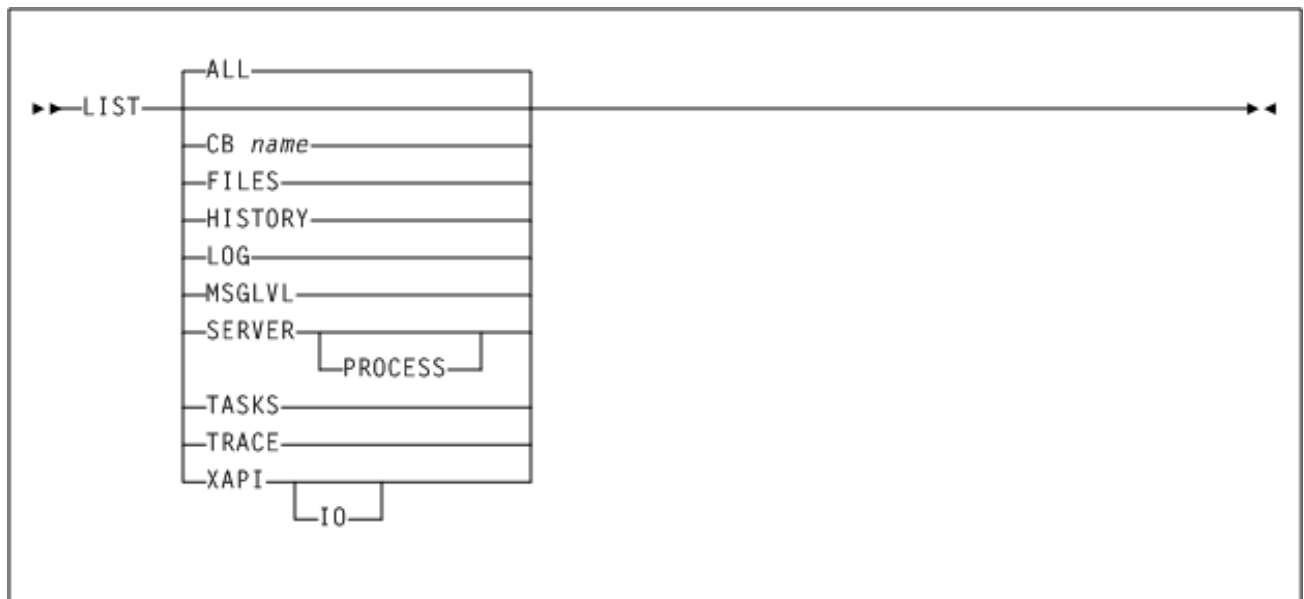
Description

La commande *LIST* répertorie le statut et les paramètres actuels du serveur XAPI.

Syntaxe

La figure suivante illustre la syntaxe de la commande *LIST* :

Figure 4.2. Syntaxe de la commande LIST



Paramètres

Comme illustré dans la [Figure 4.2, « Syntaxe de la commande LIST »](#), la commande *LIST* inclut les paramètres suivants :

ALL

facultativement, affiche tous les statuts et paramètres du serveur XAPI. Il s'agit du comportement par défaut si vous exécutez la commande *LIST* sans paramètre.

CB name

Indique que le bloc de contrôle nommé est affiché sous forme de caractères et au format hexadécimal. Uniquement pour fournir des informations au support technique Oracle.

Spécifiez l'un des blocs de contrôle suivants pour *name* :

- *HTTPCVT*
- *HTTPGBL*
- *HTTPREQ-*nnn**
- *HTTPAPI-*nnn**

Pour *LIST CB HTTPREQ* et *LIST CB HTTPAPI*, un indice entre 0 et 999 doit être spécifié. En outre, un tiret ("-") sépare l'index du nom de bloc de contrôle, sans espace intermédiaire. Par exemple :

LIST HTTPREQ-0

ou

LIST HTTPAPI-900

FILES

facultativement, indique que le nom du chemin complet des fichiers journal, trace et de contrôle est affiché.

HISTORY

facultativement, indique que l'historique du nombre de transactions XAPI des dernières 24 h est affiché.

LOG

facultativement, indique que le paramètre du journal actuel du serveur XAPI est affiché. Cela équivaut à exécuter la commande *LOG* sans paramètre.

MSGLVL

facultativement, indique que le paramètre actuel de niveau de message du serveur XAPI est affiché. Cela équivaut à exécuter la commande *MSGLVL* sans paramètre.

SERVER

facultativement, indique que le serveur XAPI et les versions UNIX actuels, les paramètres pertinents, les variables environnementales, le segment partagé et les files de messages sont affichés.

Vous pouvez facultativement inclure le mot clé *PROCESS* pour demander l'affichage de tous les ID de processus serveur XAPI, des chaînes de messages, des nombres d'ouvertures de fichier et de l'utilisation de la CPU et de la mémoire.

PROCESS

Lorsqu'il indique *SERVER*, le mot clé *PROCESS* demande aussi l'affichage de tous les ID de processus serveur XAPI actifs, des chaînes de messages, des nombres d'ouvertures de fichier et de l'utilisation de la CPU et de la mémoire.

TASKS

facultativement, indique que le système actuel du serveur XAPI et les tâches de travail sont affichés.

TRACE

facultativement, indique que le paramètre de trace actuel du serveur XAPI est affiché. Cela équivaut à exécuter la commande *TRACE* sans paramètre.

XAPI

facultativement, indique que le port d'écoute du serveur XAPI et l'adresse IP actuels sont affichés.

Vous pouvez facultativement inclure le mot clé *IO* pour demander l'affichage de toutes les statistiques du port d'écoute du serveur XAPI.

Exemple

Dans l'exemple suivant, un utilisateur exécute la commande *List* pour afficher le port d'écoute du serveur XAPI, l'adresse IP et le statut du port d'écoute actuels :

```
XCMD LIST XAPI IO
```

MSGLVL

La section suivante décrit la commande *MSGLVL*.

Description

La commande *MSGLVL* modifie ou répertorie le niveau de détail du message du serveur XAPI.

Syntaxe

La figure suivante illustre la syntaxe de la commande *MSGLVL* :

Figure 4.3. Syntaxe de la commande MSGLVL



Paramètres

Comme illustré dans la [Figure 4.3, « Syntaxe de la commande MSGLVL »](#), la commande *MSGLVL* inclut les paramètres suivants :

List

Indique que le paramètre *MSGLVL* actuel est affiché. Si aucun paramètre de commande n'est spécifié, la valeur par défaut est List.

level

indique le paramètre *MSGLVL* souhaité. Plus le niveau est élevé, plus l'envoi de messages du serveur XAPI est détaillé. Les niveaux de message sont cumulatifs ; *MSGLVL 8* produit tous les messages jusqu'à *MSGLVL 8* y compris (c'est-à-dire, de *MSGLVL 0* à *8*). Les niveaux de message sont généralisés comme suit :

- 0 - Messages de démarrage, d'arrêt et d'erreur normaux ; ces messages ne peuvent pas être supprimés.
- 4 - Messages d'avertissement sérieux.
- 8 - Messages d'avertissement secondaires.
- 12 - Messages de paramètres et d'options de démarrage.
- 16 - Messages supplémentaires de démarrage et d'arrêt.
- 20 - Messages supplémentaires de diagnostic, niveau 20.
- 24 - Messages supplémentaires de diagnostic, niveau 24.
- 16 - Messages supplémentaires de tâche de démarrage et d'arrêt.

OFF

Indique que tous les messages détaillés doivent être désactivés. Cela équivaut à *MSGLVL 0*.

Exemple

Dans l'exemple suivant, un utilisateur exécute la commande *MSGLVL* pour modifier l'envoi de messages du serveur XAPI afin d'inclure tous les messages d'avertissement :

```
XCMD MSGLVL 8
```

TRACE

La section suivante décrit la commande *TRACE*.

Description

La commande *TRACE* modifie ou répertorie les paramètres trace du serveur XAPI.

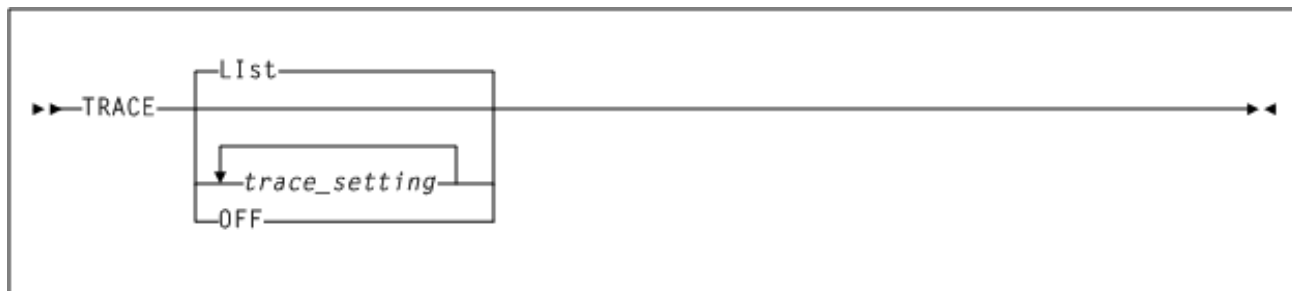
Remarque:

- Le nom et l'emplacement du fichier trace sont définis par les variables environnementales ACSLS. Par défaut, le fichier est nommé *vtrace.file* et se trouve dans la variable *DV_TAG_XAPI_WORK_PATH*.
 - La fonction de trace peut avoir un impact significatif sur les performances du système. N'activez la fonction de trace que sur demande du support d'Oracle StorageTek
 - La commande *TRACE* vous permet de définir et d'afficher plus de positions de "0" et de "1" (16) que celles définies actuellement dans les paramètres *TRACE* ci-dessus ; cela permet de futures expansions, et tout "0" ou "1" superflus du paramètre *TRACE* sera ignoré.
-

Syntaxe

La figure suivante illustre la syntaxe de la commande *TRACE* :

Figure 4.4. Syntaxe de la commande TRACE



Paramètres

Comme illustré dans la [Figure 4.4, « Syntaxe de la commande TRACE »](#), la commande *TRACE* inclut les paramètres suivants :

LIst

facultativement, indique que le paramètre trace actuel est affiché. Si aucun paramètre de commande n'est spécifié, la valeur par défaut est *LIst*.

trace_setting

facultativement, indique le paramètre trace souhaité sous la forme d'une série de "0" et de "1". 0 signifie que le paramètre trace positionnel souhaité doit correspondre à *OFF* ou être désactivé, alors que 1 signifie que le paramètre trace positionnel souhaité doit correspondre à *ON* ou être activé. La valeur spécifiée remplace complètement et n'est

pas fusionnée avec le paramètre trace actuel. Les paramètres trace positionnels sont les suivants :

- *1* - Trace les erreurs dans le fichier trace.
- *01* - Trace les fonctions et événements TCP/IP dans le fichier trace.
- *001* - Trace les fonctions et événements PGMI ou ACSAPI dans le fichier trace.
- *0001* - Trace les événements de serveur XAPI autrement non classés dans le fichier trace.
- *00001* - Trace les événements malloc() et free() dans le fichier trace.
- *000001* - Trace les événements d'analyseur XML dans le fichier trace.
- *0000001* - Trace les événements de processus serveur de commande dans le fichier trace.
- *00000001* - Trace les événements de processus de contrôle dans le fichier trace.
- *000000001* - Trace les fonctions et événements CSV dans le fichier trace.

OFF

facultativement, indique que la fonction de trace est désactivée. Cela équivaut à *TRACE 0*.

Exemple

Dans l'exemple suivant, un utilisateur exécute la commande *TRACE* pour configurer le serveur XAPI de manière à tracer toutes les erreurs et tous les événements malloc() et free() dans le fichier trace :

```
XCMD TRACE 10001
```


Chapitre 5. XCMD prend en charge les demandes d'utilisateur XAPI et les commandes HSC/VTCS

En plus des commandes de configuration et d'administration du serveur XAPI décrites dans [Chapitre 3, Commandes de configuration et d'administration XCMD XAPI](#) et des commandes d'opérateur du serveur XAPI décrites dans [Chapitre 4, Commandes d'opérateur du serveur XCMD XAPI](#), l'interface XCMD prend également en charge de nombreuses demandes d'utilisateur XAPI et les commandes de texte HSC/MVS compatibles.

Par exemple, pour saisir une demande `QUERY_VOLUME` à partir d'ACSL user_proc et de la commande XCMD :

```
XCMD QUERY_VOLSER S00001
```

Demandes, commandes et paramètres XAPI pris en charge

La table suivante répertorie les demandes XAPI et les commandes de texte HSC/MVS correspondantes qui sont prises en charge par XCMD :

Tableau 5.1. Demandes XAPI et commandes HSC/MVS correspondantes prises en charge par XCMD

Demande XAPI	Commande de texte HSC/MVS correspondante
<code>DISMOUNT</code>	<code>DISMOUNT</code>
<code>EJECT</code>	<code>EJECT</code>
<code>MOUNT</code>	<code>MOUNT</code>
<code>MOVE</code>	<code>MOVE</code>
<code>QUERY_CAP</code>	<code>Display Cap</code>
<code>QUERY_DRIVES</code>	<code>Display DRives</code>
<code>QUERY_DRIVE_INFO</code>	<code>Display DRIVE_INFO</code>
<code>QUERY_LSM</code>	<code>Display Lsm</code>
<code>QUERY_SERVER</code>	Aucune commande de texte HSC/MVS correspondante
<code>QUERY_SCRATCH</code>	<code>Display SCRATCH</code>
<code>QUERY_THRESHOLD</code>	<code>Display THRESHOLD</code>
<code>QUERY_VOLSER</code> ou <code>QUERY_VOLUME</code>	<code>Display Volser</code>
<code>QUERY_VOLUME_INFO</code>	<code>Display VOLUME_info</code>

Demande XAPI	Commande de texte HSC/MVS correspondante
<i>SCRATCH</i>	<i>SCRAtch</i>
<i>UNSCRATCH</i>	<i>UNSCratch</i>
<i>VOLRPT</i> ¹	<i>VOLRpt</i>

¹Ce *VOLRPT* correspond à la commande et au rapport *VOLRPT* au format HSC, NON au format ACSLS.

Pour plus d'informations sur les commandes de texte HSC/MVS répertoriées ci-dessus, reportez-vous à la section *Référence des commandes, des instructions de contrôle et des utilitaires ELS*.

Même si ces demandes et commandes sont prises en charge, tous les paramètres ne sont pas valides dans ACSLS. Les paramètres non valides sont comme suit :

- *DISMOUNT* (ou la commande *DISMount*)

Spécification des résultats suivants dans une erreur de syntaxe :

- Format d'adresse ccuu d'appareil MVS
- Paramètre de position hostid

- *EJECT* (ou la commande *EJect*)

Spécification des résultats des paramètres suivants dans une erreur de syntaxe :

- *WAITCAP*
- *SEQ*
- *RECTECH*

- *MOUNT* (ou la commande *Mount*)

Spécification des résultats suivants dans une erreur de syntaxe :

- Format d'adresse ccuu d'appareil MVS
- Paramètre de position hostid
- Le paramètre *MGMTCLAS* n'est pas valide

- *MOVE* (ou la commande *MOVE*)

Tous les paramètres sont pris en charge

- *QUERY_CAP* (ou la commande *Display Cap*)
 - Tous les paramètres sont pris en charge
- *QUERY_DRIVES* (ou la commande *Display DRives*)

Spécification des résultats des paramètres suivants dans une erreur de syntaxe :

- *ALL*
- *IDLE*
- *LIBRARY*
- *DETAIL*

- *IDENTITY*
- *MEDIA*
- *RECTECH*
- *UNIT*
- *BYDRIVE*
- *BYLOC*
- *SHOWLSLOT*
- *QUERY_DRIVE_INFO* (ou la commande *Display DRIVE_INFO*)

Spécification des résultats des paramètres suivants dans une erreur de syntaxe :

- *ALL*
- *IDLE*
- *LIBRARY*
- *DETAIL*
- *IDENTITY*
- *MEDIA*
- *RECTECH*
- *VIRTUAL*
- *DEVADDR*
- *LSMLOC*
- *QUERY_LSM* (ou la commande *Display Lsm*)

Tous les paramètres sont pris en charge.

- *QUERY_SERVER* (ou la commande *Display SERVER*)

Tous les paramètres sont pris en charge.

- *QUERY_SCRATCH* (ou la commande *Display SCRatch*)

Tous les paramètres sont pris en charge.

- *QUERY_THRESHOLD* (ou la commande *Display THReshold*)

Tous les paramètres sont pris en charge.

- *QUERY_VOLSER* (ou la commande *Display Volser*)

Tous les paramètres sont pris en charge.

- *QUERY_VOLUME_INFO* (ou la commande *Display VOLume_info*)

Tous les paramètres sont pris en charge.

- *SCRATCH* (ou la commande *SCRATch*)

Tous les paramètres sont pris en charge.

- *UNSCRATCH* (ou la commande *UNSCRatch*)

Tous les paramètres sont pris en charge.

- *VOLRPT* (ou la commande *VOLRpt*)

Spécification des résultats suivants dans une erreur de syntaxe :

- *MEDEQUAL*
- *NONMEDEQ*
- *VAULT*

Exécution de commandes vers le serveur ACSLS XAPI avec la commande Route d'un client SMC ou VM

Exception faite des commandes *EJECT* et *VOLRpt*, vous pouvez exécuter toute commande HSC/MVS répertoriée dans [Tableau 5.1, « Demandes XAPI et commandes HSC/MVS correspondantes prises en charge par XCMD »](#) à partir d'un client SMC/MVS ou d'un client VM vers le serveur ACSLS XAPI avec la commande *Route*.

Contrairement aux commandes *XCMD* saisies à partir d'ACSLs cmd_proc ou d'un client SMC/MVS ou VM, la commande *Route* est conforme aux conventions syntaxiques SMC/HSC. Aussi, le jeton *XCMD* est omis.

Par exemple :

Pour exécuter une demande *QUERY VOLUME XAPI* sous la forme d'une commande ACSLS cmd_proc *XCMD* :

```
XCMD QUERY VOLUME VOLSER
```

Où *VOLSER* correspond au volume souhaité.

Pour exécuter la demande *QUERY VOLUME XAPI* à partir d'un client SMC/MVS vers le serveur ACSLS XAPI avec la commande *ROUTE* :

```
ROUTE ACSLS, QUERY VOLUME(volser)
```

Où *volser* correspond au volume souhaité.

Reportez-vous à la *Référence des commandes, des instructions de contrôle et des utilitaires ELS* pour plus d'informations sur la commande *ROUTE* du SMC.

Chapitre 6. Variables de contrôle du serveur XAPI ACSLS

Le logiciel ACSLS d'Oracle fournit un ensemble de variables du système qui vous permettent de contrôler certains comportements de votre système ACSLS. Plusieurs variables de contrôle propres au composant du serveur ACSLS XAPI sont incluses. Ces variables comprennent :

- [XAPI_PORT](#)
- [XAPI_WORK_PATH](#)
- [XAPI_LOG_SIZE](#)
- [XAPI_LOG_FILE_NUM](#)
- [XAPI_TRACE_SIZE](#)
- [XAPI_TRACE_FILE_NUM](#)
- [XAPI_STARTUP_FILE](#)
- [XAPI_TAPEPLEX_NAME](#)

Configuration des variables de contrôle du serveur XAPI ACSLS

Utilisez l'utilitaire *acsss_config* ou *dv_config* d'ACSLs pour afficher et définir les variables statiques XAPI ACSLS. Vous devez redémarrer ACSLS pour que les modifications soient appliquées.

- Pour utiliser l'utilitaire *dv_config*, entrez l'une des commandes suivantes :
 - *dv_config -d* pour afficher toutes les variables ACSLS.
 - *dv_config -p <variable_name>* pour mettre à jour la variable XAPI.
- Pour utiliser l'utilitaire *acsss_config*, utilisez le script *acsss_config* d'ACSLs pour accéder à l'écran ACSLS Feature Configuration :

ACSLs Feature Configuration

Please enter the number followed by Return for your choice
from the following menu to configure product behavior in that area.

Press ? followed by the Return key for help.

- 1: Set CSI tuning variables
- 2: Set event logging variables
- 3: Set general product behavior variables
- 4: Set access control variables
- 5: Set automatic backup parameters
- 6: Rebuild Access Control information
- 7: Event Notification settings
- 8: Define or Change Library Configuration
- 9: Set XAPI server variables
- E: Exit

Sélectionnez l'option 9 (*Set XAPI server variables*).

Remarque:

Cette option n'est visible que si la fonction du serveur ACSLS XAPI est activée.

Reportez-vous au *Guide de l'administrateur de StorageTek Automated Cartridge System Library Software* pour obtenir plus d'informations sur les utilitaires *acsss_config* et *dv_config*, et sur la façon d'utiliser ces utilitaires pour afficher et mettre à jour les variables d'ACSLs.

Description de variable ACSLS XAPI

Cette section décrit les variables valides utilisées pour contrôler le composant du serveur ACSLS XAPI.

XAPI_PORT

Invite : *Changes to the user-defined inbound port to the XAPI server will not take effect until the XAPI server is restarted. Port number used by the XAPI server to receive incoming XAPI requests. [50020]:*

Cette option spécifie le port utilisé par le serveur XAPI pour les demandes TCP entrantes émises par les clients. Saisissez un nombre entre 1024 et 65535 afin de définir le port utilisé par le serveur XAPI. Ne spécifiez pas le port 50003. Reportez-vous au *Guide de sécurité ACSLS Security* pour plus d'informations.

Le serveur XAPI doit être redémarré pour que cette variable soit appliquée.

XAPI_WORK_PATH

Invite : *Changes to the XAPI work directory will not take effect until the xapi server is restarted. Place the XAPI log and trace information in which directory. [\$ACS_HOME/log/xapi]*

Sélectionnez le répertoire où les fichiers de travail du serveur XAPI seront placés. Une fois installé, le serveur XAPI journalise les informations dans le répertoire `$ACS_HOME/Log/xapi`. Dans des conditions d'utilisation normales, la valeur de cette variable n'est pas modifiée. En cas de problèmes liés à l'espace disque dans le système de fichiers contenant `$ACS_HOME`, un autre chemin peut être spécifié. Le chemin doit être absolu (par exemple, un lien qui commence par `/` ou `$ACS_HOME`). Le serveur XAPI doit être redémarré pour que cette variable soit appliquée.

XAPI_LOG_SIZE

Invite : *Changes to the XAPI log size will not take effect until the XAPI server is restarted. Maximum XAPI log size in Mbytes [20]:*

Cette option spécifie la taille limite du journal XAPI, elle est exprimée en mégaoctets (correspondant ici à "1048576 octets."). Saisissez un nombre non négatif. 20 correspond à la valeur par défaut pour cette option.

Le serveur XAPI doit être redémarré pour que cette variable soit appliquée.

XAPI_LOG_FILE_NUM

Invite : *Number of XAPI Log archive files to retain [10]:*

Cette option spécifie le nombre de fichiers journaux XAPI archivés à conserver.

Lorsque la taille actuelle du fichier `vlog.file` dépasse la taille limite, le fichier journal est renommé avec un suffixe allant de 0 à *n*. 0 correspond au fichier le plus récent et *n* au plus ancien. Les fichiers archivés sont enregistrés dans le répertoire `API_WORK_PATH`.

Une fois le nombre spécifié de journaux archivés atteint, le fichier le plus ancien est supprimé du répertoire d'archivage chaque fois qu'un nouveau fichier y est ajouté. Vous ne pouvez pas conserver moins d'un fichier archivé et pas plus de 99.

Saisissez un nombre entre 1 et 99 pour spécifier le nombre de fichiers journaux archivés à conserver.

Le serveur XAPI doit être redémarré pour que cette variable soit appliquée.

XAPI_TRACE_SIZE

Invite : *Changes to the XAPI trace size will not take effect until the xapi server is restarted. Taille maximale de la trace XAPI en mégaoctets. [50] :*

Cette option spécifie la taille limite de la trace XAPI, elle est exprimée en mégaoctets (correspondant ici à "1048576 octets."). Saisissez un nombre non négatif. 50 correspond à la valeur par défaut pour cette option.

Le serveur XAPI doit être redémarré pour que cette variable soit appliquée.

XAPI_TRACE_FILE_NUM

Invite : *Number of XAPI Trace archive files to retain [10]:*

Cette option indique le nombre de fichiers XAPI TRACE archivés à conserver. Lorsque la taille actuelle du fichier trace.file dépasse la taille limite, le fichier trace est renommé avec un suffixe allant de 0 à *n*. 0 correspond au fichier le plus récent et *n* au plus ancien. Les fichiers archivés sont enregistrés dans le répertoire *XAPI_WORK_PATH*. Une fois le nombre spécifié de journaux archivés atteint, le fichier le plus ancien est supprimé du répertoire d'archivage chaque fois qu'un nouveau fichier y est ajouté. Vous pouvez conserver de 1 à 99 fichiers archivés. Saisissez un nombre entre 1 et 99 pour spécifier le nombre de fichiers journaux archivés à conserver.

Le serveur XAPI doit être redémarré pour que cette variable soit appliquée.

XAPI_STARTUP_FILE

Invite : *Changes to the XAPI startup file name will not take effect until the xapi server is restarted. Name of the XAPI startup file with control parms [xapi_startup_file]*

Cette option spécifie le nom du fichier de démarrage XPI. Ce fichier réside dans le répertoire *XAPI_WORK_PATH* et inclut les paramètres de démarrage XAPI.

Le serveur XAPI doit être redémarré pour que cette variable soit appliquée.

XAPI_TAPEPLEX_NAME

Invite : *Changes to the XAPI Tapeplex name will not take effect until the xapi server is restarted. Name of the XAPI Tapeplex []:*

Cette option spécifie le nom du Tapeplex XAPI. Saisissez un texte ne contenant pas plus de 8 caractères. Le serveur XAPI doit être redémarré pour que cette variable soit appliquée.

Index

C

Commande de configuration, 21
Commande DEFINE POOL_Access, 22
Commande DEFINE POOL_Name, 21
Commande DELETE POOL_Access, 25
Commande DELETE POOL_Name, 24
Commande LIST, 35
Commande LOG, 33
Commande MSGLVL, 37
Commande QUERY POOL_Access, 30
Commande QUERY POOL_Name, 29
Commande SET POOL_Name, 26
Commande TRACE, 39
Commandes
 Commandes d'opérateur du serveur XAPI, 33
 Commandes de configuration et d'administration XAPI, 21
 DEFINE POOL_Name, 21
 DEFINE POOL_Access, 22
 DELETE POOL_Name, 24
 DELETE POOL_Access, 25
 LIST, 35
 LOG, 33
 MSGLVL, 37
 QUERY POOL_Name, 29
 QUERY POOL_Access, 30
 SET POOL_Name, 26
 TRACE, 39
Commandes d'administration, 21
Commandes d'opérateur, 33
conventions des commandes, 19
conventions, commande, 19

D

description
 Composant d'interface XCMD, 17
 Interface client XAPI pour serveur ACSLS, 17

E

Exemples
 Commande DELETE POOL_Name, 25
 Commande DELETE POOL_Access, 26

Commande LOG, 35, 37
Commande MSGLVL, 38
Commande QUERY POOL_Name, 29
Commande QUERY POOL_Access, 31
Commande Route, 44
Commande TRACE, 40
DEFINE POOL_Access, 22, 24
SET POOL_Name, 28

I

Introduction, 17

S

saisie de commandes
 à partir d'ACSLS cmd-proc, 19
 avec la commande Route d'un client SMC ou VM, 44
syntaxe
 Commande DEFINE POOL_Name, 21
 Commande DEFINE POOL_Access, 23
 Commande DELETE POOL_Name, 24
 Commande DELETE POOL_Access, 25
 Commande LOG, 33, 35
 Commande MSGLVL, 37
 Commande QUERY POOL_Name, 29
 Commande QUERY POOL_Access, 30
 Commande SET POOL_Name, 27
 Commande TRACE, 39

V

Variables de contrôle XAPI ACSLS, 45
variables, ACSLS, 45
