

StorageTek Enterprise Library Software

Referencia de la interfaz del cliente XAPI para el servidor ACSLS

E74278-01

Septiembre de 2015

StorageTek Enterprise Library Software

Referencia de la interfaz del cliente XAPI para el servidor ACSLS

E74278-01

Copyright © 2015, Oracle y/o sus filiales. Todos los derechos reservados.

Este software y la documentación relacionada están sujetos a un contrato de licencia que incluye restricciones de uso y revelación, y se encuentran protegidos por la legislación sobre la propiedad intelectual. A menos que figure explícitamente en el contrato de licencia o esté permitido por la ley, no se podrá utilizar, copiar, reproducir, traducir, emitir, modificar, conceder licencias, transmitir, distribuir, exhibir, representar, publicar ni mostrar ninguna parte, de ninguna forma, por ningún medio. Queda prohibida la ingeniería inversa, desensamblaje o descompilación de este software, excepto en la medida en que sean necesarios para conseguir interoperabilidad según lo especificado por la legislación aplicable.

La información contenida en este documento puede someterse a modificaciones sin previo aviso y no se garantiza que se encuentre exenta de errores. Si detecta algún error, le agradeceremos que nos lo comunique por escrito.

Si este software o la documentación relacionada se entrega al Gobierno de EE.UU. o a cualquier entidad que adquiera las licencias en nombre del Gobierno de EE.UU. entonces aplicará la siguiente disposición:

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.

Este software o hardware se ha desarrollado para uso general en diversas aplicaciones de gestión de la información. No se ha diseñado ni está destinado para utilizarse en aplicaciones de riesgo inherente, incluidas las aplicaciones que pueden causar daños personales. Si utiliza este software o hardware en aplicaciones de riesgo, usted será responsable de tomar todas las medidas apropiadas de prevención de fallos, copia de seguridad, redundancia o de cualquier otro tipo para garantizar la seguridad en el uso de este software o hardware. Oracle Corporation y sus filiales declinan toda responsabilidad derivada de los daños causados por el uso de este software o hardware en aplicaciones de riesgo.

Oracle y Java son marcas comerciales registradas de Oracle y/o sus filiales. Todos los demás nombres pueden ser marcas comerciales de sus respectivos propietarios.

Intel e Intel Xeon son marcas comerciales o marcas comerciales registradas de Intel Corporation. Todas las marcas comerciales de SPARC se utilizan con licencia y son marcas comerciales o marcas comerciales registradas de SPARC International, Inc. AMD, Opteron, el logotipo de AMD y el logotipo de AMD Opteron son marcas comerciales o marcas comerciales registradas de Advanced Micro Devices. UNIX es una marca comercial registrada de The Open Group.

Este software o hardware y la documentación pueden proporcionar acceso a, o información sobre contenidos, productos o servicios de terceros. Oracle Corporation o sus filiales no son responsables y por ende desconocen cualquier tipo de garantía sobre el contenido, los productos o los servicios de terceros a menos que se indique otra cosa en un acuerdo en vigor formalizado entre Ud. y Oracle. Oracle Corporation y sus filiales no serán responsables frente a cualesquiera pérdidas, costos o daños en los que se incurra como consecuencia de su acceso o su uso de contenidos, productos o servicios de terceros a menos que se indique otra cosa en un acuerdo en vigor formalizado entre Ud. y Oracle.

Tabla de contenidos

Prefacio	9
Destinatarios	9
Accesibilidad a la documentación	9
Documentos relacionados	9
Convenciones	9
Convenciones de sintaxis	10
Líneas de flujo	10
Elección única obligatoria	10
Elección única opcional	10
Valores por defecto	11
Repetición	11
Palabras clave	11
Variables	11
Valores alternativos	12
Opcional	12
Delimitadores	12
Rangos	12
Rangos hexadecimales	12
Rangos decimales	12
Rangos de volers con formato HSC	12
Rangos de volers alfanuméricos con formato ACSLS	14
Listas	14
Espacios en blanco	15
1. Introducción	17
¿Qué es la interfaz del cliente XAPI para el servidor ACSLS?	17
¿Qué es el componente de interfaz XCMD?	17
2. Convenciones de comandos XCMD	19
3. Comandos de configuración y administración de XAPI de XCMD	21
DEFINE POOL_Name	21
Descripción	21

Sintaxis	21
Parámetros	22
Ejemplo	22
DEFINE POOL_Access	22
Descripción	22
Sintaxis	23
Parámetros	23
Ejemplo	24
DELETE POOL_Name	24
Descripción	24
Sintaxis	24
Parámetros	25
Ejemplo	25
DELETE POOL_Access	25
Descripción	25
Sintaxis	26
Parámetros	26
Ejemplo	26
SET POOL_Name	26
Descripción	27
Sintaxis	27
Parámetros	27
Ejemplo	29
QUERY POOL_Name	29
Descripción	29
Sintaxis	29
Parámetros	29
Ejemplo	30
QUERY POOL_Access	30
Descripción	30
Sintaxis	30
Parámetros	31
Ejemplo	31
4. Comandos del operador del servidor de la XAPI de XCMD	33
LOG	33
Descripción	33
Sintaxis	33
Parámetros	34

Ejemplo	35
LIST	35
Descripción	35
Sintaxis	35
Parámetros	35
Ejemplo	37
MSGLVL	37
Descripción	37
Sintaxis	37
Parámetros	38
Ejemplo	38
TRACE	38
Descripción	38
Sintaxis	39
Parámetros	39
Ejemplo	40
5. Compatibilidad de XCMD con solicitudes de usuarios de XAPI y comandos de HSC/VTCS	41
Solicitudes, comandos y parámetros de XAPI admitidos	41
Ejecución de comandos para el servidor de la XAPI de ACSLS utilizando el comando route del cliente de VM o SMC	44
6. Variables de control del servidor de la XAPI de ACSLS	45
Configuración de variables de control del servidor de la XAPI de ACSLS	45
Descripciones de las variables de XAPI de ACSLS	46
XAPI_PORT	46
XAPI_WORK_PATH	47
XAPI_LOG_SIZE	47
XAPI_LOG_FILE_NUM	47
XAPI_TRACE_SIZE	48
XAPI_TRACE_FILE_NUM	48
XAPI_STARTUP_FILE	48
XAPI_TAPEPLEX_NAME	48
Índice	51

Lista de tablas

5.1. Solicitudes de XAPI y comandos de HSC/MVS correspondientes admitidos por XCMD	41
--	----

Prefacio

En esta publicación, se describe cómo configurar y administrar un servidor Oracle StorageTek ACSLS con la compatibilidad con XAPI instalada.

Destinatarios

Este documento está destinado a administradores de almacenamiento, programadores de sistemas y operadores responsables de la configuración y el mantenimiento del servidor de XAPI de ACSLS.

Accesibilidad a la documentación

Para obtener información sobre el compromiso de Oracle con la accesibilidad, visite el sitio web del Programa de Accesibilidad de Oracle en <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Acceso a My Oracle Support

Los clientes de Oracle que hayan contratado servicios de soporte electrónico pueden acceder a ellos mediante My Oracle Support. Para obtener información, visite <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> o, si tiene alguna discapacidad auditiva, visite <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>.

Documentos relacionados

Para obtener más información, consulte los siguientes documentos del juego de documentación de Oracle StorageTek ELS versión 7.3 y del juego de documentación de ACSLS versión 8.4:

- *Guía del administrador de ACSLS 8.4*
- *Guía de instalación de ACSLS 8.4*
- *Referencia rápida de ACSLS 8.4*
- *Referencia de comandos, sentencias de control y utilidades de ELS 7.3*
- *Configuración y gestión de SMC 7.3*
- *Guía de instalación, configuración y administración del cliente de VM*

Convenciones

Se han utilizado las siguientes convenciones en este documento:

Convención	Significado
negrita	El tipo de fuente en negrita indica elementos de la interfaz gráfica de usuario asociados a una acción o términos definidos en el texto o el glosario.

Convención	Significado
<i>cursiva</i>	El formato de cursiva indica títulos de libros, énfasis o variables de pendientes de asignación para los que se proporcionan valores concretos.
<i>monoespaciado</i>	El formato de monoespacio indica comandos en un párrafo, direcciones URL, código en los ejemplos, texto que aparece en la pantalla o texto que se introduce.

Convenciones de sintaxis

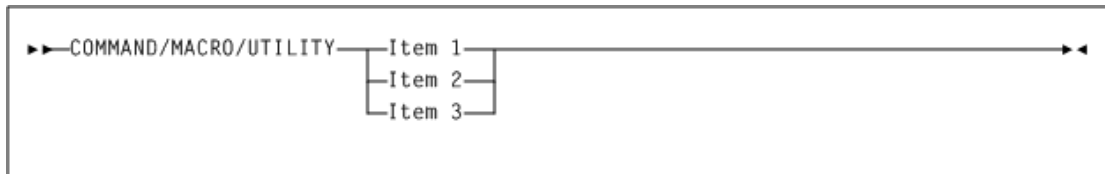
Nota:

Las siguientes convenciones de sintaxis se aplican **solo** a la sintaxis de XAPI y XCMD. Consulte la *Guía del administrador de ACSLS* para obtener información sobre las convenciones de sintaxis que se aplican a los comandos y utilidades de ACSLS.

Las convenciones de diagrama de flujo de sintaxis incluyen lo siguiente:

Líneas de flujo

Los diagramas de sintaxis están compuestos por una línea de base horizontal, líneas de ramificación horizontales y verticales, y el texto correspondiente a un comando, sentencia de control, macro o utilidad. Los diagramas se leen de izquierda a derecha, y de arriba hacia abajo. Las flechas indican flujo y dirección. Por ejemplo:



Elección única obligatoria

Las líneas de ramificación indican que se debe elegir una opción. Si uno de los elementos entre los cuales se puede elegir se encuentra en la línea de base del diagrama, se debe seleccionar un elemento. Por ejemplo:



Elección única opcional

Si el primer elemento se encuentra debajo de la línea de base, se puede seleccionar un elemento de manera opcional. Por ejemplo:



Valores por defecto

Los parámetros y valores por defecto aparecen por encima de la línea de base. Por ejemplo:

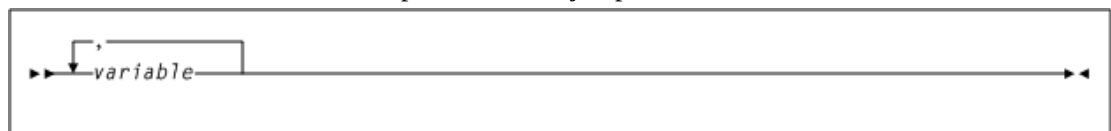


Algunos parámetros de palabras claves permiten elegir valores de una pila. Cuando la pila contiene un valor por defecto, la palabra clave y las opciones de valores aparecen debajo de la línea de base para indicar que son opcionales y el valor por defecto aparece por encima de la línea de la palabra clave. Por ejemplo:



Repetición

Un símbolo de repetición indica que se puede seleccionar más de una opción o que se puede seleccionar una única opción más de una vez. El siguiente ejemplo indica que se debe utilizar una coma como delimitador de repetición. Por ejemplo:



Palabras clave

Todas las palabras clave de comandos se muestran en mayúsculas o en una combinación de mayúsculas y minúsculas. Cuando los comandos no distinguen entre mayúsculas y minúsculas, una combinación de ambas letras implica que las minúsculas pueden omitirse para formar una abreviación.

Variables

El formato de cursiva indica una variable.

Valores alternativos

La barra vertical (|) separa valores de parámetros alternativos.

Opcional

Los corchetes [] indican que un parámetro de comando es opcional.

Delimitadores

Si una coma (,), un punto y coma (;) u otro delimitador se muestra con un elemento del diagrama de sintaxis, se debe escribir como parte de la sentencia.

Rangos

Un par de elementos con la misma longitud y el mismo tipo de datos, unidos por un guion, indica un rango inclusivo. El primer elemento debe ser estrictamente menor que el segundo elemento.

Rangos hexadecimales

Un rango hexadecimal consta de un par de números hexadecimales (por ejemplo, 0A2-0AD o 000-0FC).

Rangos decimales

Un rango decimal consta de un par de números decimales (por ejemplo, 1-9 o 010-094). No son necesarios los ceros iniciales. La parte decimal es un rango incremental. Las posiciones de los caracteres correspondientes a la parte incremental de ambos elementos del rango deben coincidir y los caracteres no incrementales del primer elemento deben ser idénticos a los del segundo elemento.

Rangos de volsers con formato HSC

Un rango de volsers (vol-range) numérico con formato HSC consta de un par de elementos VOLSER que contienen una parte numérica decimal de 1 a 6 dígitos (por ejemplo, ABC012-ABC025 o X123CB-X277CB). La parte decimal es un rango incremental. Se aplican las siguientes restricciones adicionales:

- Las posiciones de los caracteres correspondientes a la parte incremental de ambos elementos del rango deben coincidir.
- Los caracteres no incrementales del primer elemento deben ser idénticos a los del segundo elemento.
- No puede incrementar dos partes de un elemento del rango. Si 111AAA es el primer elemento, no puede especificar 112AAB para el segundo elemento.
- Si un rango VOLSER contiene más de una parte decimal, cualquier parte es válida como rango incremental. Por ejemplo:

- A00B00: el mayor rango que puede especificarse es de A00B00 a A99B99.
- A0B0CC: el mayor rango que puede especificarse es de A0B0CC a A9B9CC.
- 000XXX: el mayor rango que puede especificarse es de 000XXX a 999XXX.

Un rango de volsers (vol-range) alfabético con formato HSC consta de un par de elementos VOLSER que contienen una parte incremental de 1 a 6 caracteres (por ejemplo, 000AAA-000ZZZ o 9AAA55-9ZZZ55). Esta parte es un rango incremental. Se aplican las siguientes restricciones adicionales:

- Las posiciones de los caracteres correspondientes a la parte incremental de ambos elementos del rango deben coincidir.
- Los caracteres no incrementales del primer elemento deben ser idénticos a los del segundo elemento.
- No puede incrementar dos partes de un elemento del rango. Si 111AAA es el primer elemento, no puede especificar 112AAB para el segundo elemento.
- La parte alfabética del rango VOLSER se puede definir desde el carácter A al carácter Z. Para incrementar secuencias de varios caracteres, cada carácter se puede incrementar hasta Z. Por ejemplo, ACZ es parte del rango AAA-AMM. Los ejemplos son:

- A00A0-A99A0

Incrementa los VOLSER de A00A0 a A09A0 y, a continuación, de A10A0 a A99A0.

- 9AA9A-9ZZ9A

Incrementa los VOLSER de 9AA9A a 9AZ9A y, a continuación, de 9BA9A a 9ZZ9A.

- 111AAA-111ZZZ

Incrementa los VOLSER de 111AAA a 111AAZ y, a continuación, de 111ABA a 111ZZZ.

- 999AM8-999CM8

Incrementa los VOLSER de 999AM8 a 999AZ8 y, a continuación, de 999BA8 a 999CM8.

- A3BZZ9-A3CDE9

Incrementa los VOLSER de A3BZZ9 a A3CAA9 y, a continuación, de A3CAB9 a A3CDE9.

- AAAAAA-AAACCC

Incrementa los VOLSER de AAAAAA a AAAAAZ y, a continuación, de AAAABA a AAACCC.

- CCCN NN-DDDNNN

Incrementa los VOLSER de CCCNNN a CCCNNZ y, a continuación, de CCCNOA a DDDNNN. Se trata de un rango muy amplio.

El número de volúmenes de un rango de VOLSER alfabético depende del número de elementos en la parte incremental del rango de VOLSER. Para un rango de A a Z en cada carácter de posición, el número de volúmenes se puede calcular elevando 26 a la potencia del número de posiciones que se incrementan.

- A-Z es equivalente a 26^1 o 26 volúmenes.
- AA-ZZ es equivalente a 26^2 o 676 volúmenes.
- AAA-ZZZ es equivalente a 26^3 o 17.576 volúmenes.
- AAAA-ZZZZ es equivalente a 26^4 o 456.976 volúmenes.
- AAAAA-ZZZZZ es equivalente a 26^5 o 11.881.376 volúmenes.
- AAAAAA-ZZZZZZ es equivalente a 26^6 o 308.915.776 volúmenes.

Rangos de volsers alfanuméricos con formato ACSLS

No es necesario que un rango de volsers alfanumérico con formato ACSLS cumpla con los rangos de patrones de volumen con formato HSC especificados anteriormente. Se aplican las siguientes restricciones adicionales:

- Ambos volsers especificados en el rango deben contener el mismo número de caracteres.
- El primer elemento de volser en el rango debe ser estrictamente menor que el segundo elemento de volser.
- Si los volsers especificados tienen menos de 6 caracteres de longitud, los volsers se evalúan como si fueran campos de 6 caracteres en blanco y alineados a la izquierda.
- En el rango solo hay volsers estrictamente mayores o iguales que el primer elemento de volser y estrictamente menores o iguales que el segundo elemento de volser cuando se evalúan utilizando la secuencia de ordenación ASCII. Por ejemplo, el rango de volsers 123456-234567 contiene el volser 12345A.

Nota:

- Un rango de volsers con formato HSC válido es siempre válido según las reglas de volsers con formato ACSLS. Sin embargo, un rango de volsers con formato ACSLS válido puede no ser válido cuando se lo evalúa según las reglas de volsers con formato HSC.
 - La mayoría de los comandos XCMD que aceptan rangos de volsers requieren que los rangos cumplan con las reglas de volsers HSC. El único comando que acepta rangos de volsers de cualquier formato es el comando XCMD SET POOL_Name.
-

Listas

Una lista está compuesta por uno o varios elementos. Si hay más de un elemento especificado, los elementos deben estar separados por una coma o un espacio en blanco, y toda la lista debe estar delimitada por paréntesis.

Espacios en blanco

Los valores y parámetros de las palabras clave pueden estar separados por cualquier número de espacios en blanco.

Capítulo 1. Introducción

En este capítulo, se presenta la interfaz de cliente XAPI para ACSLS y se describe la interfaz de XCMD que le permite introducir comandos para un servidor ACSLS con XAPI.

¿Qué es la interfaz del cliente XAPI para el servidor ACSLS?

La API de XML (XAPI) es una API de Oracle StorageTek que permite a los clientes y los servidores de StorageTek comunicarse con un protocolo común mediante TCP/IP.

Con la presentación de esta XAPI, los clientes que anteriormente requerían el uso de un servidor basado en MVS (componente de software de host de Oracle StorageTek) para el procesamiento de cinta real ahora pueden usar ACSLS (con la compatibilidad con XAPI activada) de la siguiente manera:

- Un cliente SMC del MVS ahora puede realizar solicitudes de cinta real desde un servidor ACSLS con la compatibilidad con XAPI activada (sin necesidad de MVS/CSC).

Para obtener más información, consulte la publicación de ELS *Configuración y gestión del SMC*.

- El cliente de VM ahora puede solicitar servicios de cinta real desde un servidor ACSLS con la compatibilidad con XAPI activada.

Para obtener más información, consulte la publicación de ELS *Guía de instalación, configuración y administración del cliente de VM*.

El servicio de XAPI es compatible con ACSLS 8.4 y versiones posteriores. Consulte la *Guía de instalación de StorageTek ACSLS 8.4* para obtener más información sobre la instalación y configuración del servicio de XAPI de ACSLS.

¿Qué es el componente de interfaz XCMD?

El componente cmd_proc de ACSLS es una interfaz de línea de comandos para ACSLS que permite a los clientes gestionar un sistema ACSLS y las bibliotecas que controla ACSLS.

Consulte el capítulo Visión general de la *Guía del administrador de StorageTek Automated Cartridge System Library Software* para obtener información más detallada sobre el uso de cmd_proc. El comando XCMD es una extensión de cmd_proc de ACSLS que permite a los usuarios del componente de servidor de XAPI de ACSLS introducir un número limitado de

solicitudes de administrador y de usuario del servidor de la XAPI desde el componente `cmd_proc` de ACSLS existente cuando el componente de servidor de la XAPI de ACSLS está instalado y activo.

Consulte el [Capítulo 3, Comandos de configuración y administración de XAPI de XCMD](#) para obtener información sobre los comandos del servidor de XAPI que se pueden introducir desde `cmd_proc` de ACSLS utilizando el comando *XCMD*.

Consulte el [Capítulo 4, Comandos del operador del servidor de la XAPI de XCMD](#) para obtener información sobre los comandos del operador de XAPI que se pueden introducir desde `cmd_proc` de ACSLS utilizando el comando *XCMD*.

Además, puede usar el comando *Route* de SMC/MVS o el comando *Route* del cliente de VM para ejecutar comandos de operador del servidor de la XAPI, comandos de administración y configuración de XCMD, y comandos de XAPI de HSC/VTCS desde un host de VM o MVS. Sin embargo, se aplican ciertas restricciones de parámetros.

Consulte el [Capítulo 5, Compatibilidad de XCMD con solicitudes de usuarios de XAPI y comandos de HSC/VTCS](#) para obtener más información sobre los comandos de HSC/VTCS admitidos, otras solicitudes de XAPI y las restricciones de parámetros.

Capítulo 2. Convenciones de comandos XCMD

La mayoría de las interacciones cliente/servidor entre y clientes de VM y SMC/MVS, y un servidor ACSLS con XAPI son transparentes para el usuario final. Los clientes de VM y SMC/MVS generan las solicitudes de información de volumen, montajes y desmontajes automáticamente, y estas se procesan sin intervención del operador.

Además de estas interacciones automáticas, el servidor de ACSLS con XAPI proporciona comandos adicionales de administrador, configuración y operador que permiten gestionar el componente de XAPI.

- Consulte el [Capítulo 3, Comandos de configuración y administración de XAPI de XCMD](#) para obtener información acerca de los comandos de administración y configuración del servidor de la XAPI de XCMD.
- Consulte el [Capítulo 4, Comandos del operador del servidor de la XAPI de XCMD](#) para obtener información acerca de los comandos del operador del servidor de la XAPI de XCMD.

Estos comandos no forman parte de la instalación base de ACSLS, pero están disponibles cuando se instala opcionalmente el componente de XAPI de ACSLS. Consulte la *Guía de instalación de StorageTek Automated Cartridge System Library (ACSL)* para obtener información sobre la instalación de ACSLS y el componente de XAPI de ACSLS.

Para ejecutar estos comandos, use la utilidad de comandos XCMD, que se ejecuta desde `cmd_proc` de ACSLS. La sintaxis general de este comando es:

XCMD command

Donde *command* es uno de los comandos del componente de XAPI de ACSLS descrito en esta publicación.

Nota:

También puede usar la utilidad de comandos XCMD para ejecutar cualquier solicitud de usuario de XAPI admitida por el servidor de la XAPI de ACSLS. Consulte el [Capítulo 5, Compatibilidad de XCMD con solicitudes de usuarios de XAPI y comandos de HSC/VTCS](#) para obtener más información.

Las siguientes convenciones se aplican cada vez que ejecuta un comando XCMD desde `cmd_proc` de ACSLS:

- XCMD no usa el símbolo del sistema de ACSLS.

-
- Cuando se introducen parámetros de palabra clave con un valor asociado, el valor debe estar precedido por un solo carácter en blanco.
 - Puede introducir comandos XCMD en mayúsculas o minúsculas. El componente de XAPI de ACSLS traduce automáticamente todas las entradas de comandos a mayúsculas. Por ejemplo, los nombres de subagrupaciones "Testpool1", "testpool1", "TestPool1" y "TESTPOOL1" se traducen todos al mismo nombre de subagrupación "TESTPOOL1".

Nota:

Los comandos y parámetros normales de cmd_proc de ACSLS generalmente se especifican en minúsculas. Los caracteres en minúsculas no se traducen automáticamente a mayúsculas, excepto vol_ids (volsers).

-
- No se puede anteponer un signo igual (=) a un valor de parámetro de XCMD.
 - No se pueden encerrar los valores de XCMD entre paréntesis. Cuando se admiten valores múltiples, puede usar una coma para separar cada valor. Por ejemplo, para consultar varios volúmenes, introduzca XCMD QUERY VOLUME volser1,volser2 y no XCMD QUERY VOLUME(volser1,volser2).

Capítulo 3. Comandos de configuración y administración de XAPI de XCMD

En este capítulo, se describen los comandos de configuración y administración de XAPI admitidos por XCMD. Estos comandos activan las funciones de agrupación de volúmenes compatibles con ELS en ACSLS versión 8.4. Los comandos admitidos incluyen:

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

Ejecute estos comandos desde `user_proc` de ACSLS utilizando el comando XCMD. Consulte el [Capítulo 2, Convenciones de comandos XCMD](#) para obtener más información.

Consulte "[Convenciones de sintaxis](#)" para obtener información sobre las convenciones utilizadas en los diagramas de flujo de sintaxis incluidos en las secciones de comandos a continuación.

DEFINE POOL_Name

En la siguiente sección, se describe el comando `DEFINE POOL_Name`.

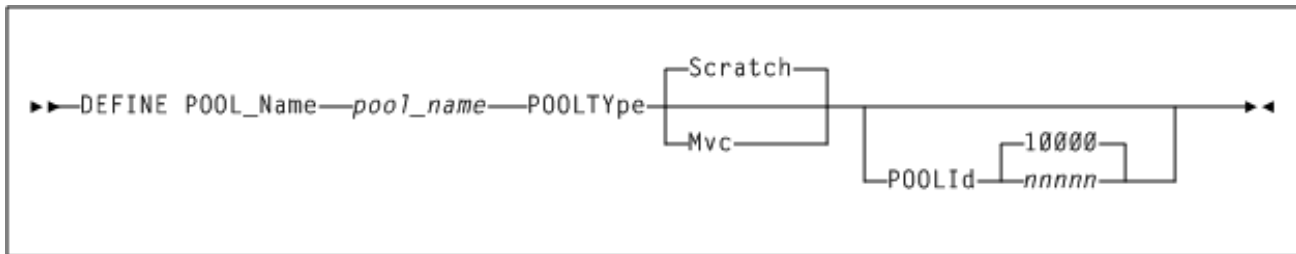
Descripción

El comando `DEFINE POOL_NAME` asocia un tipo y un nombre de agrupación con un ID de agrupación de ACSLS.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando `DEFINE POOL_Name`:

Figura 3.1. Sintaxis del comando DEFINE POOL_Name



Parámetros

Como se muestra en la [Figura 3.1, “Sintaxis del comando DEFINE POOL_Name”](#), el comando `DEFINE POOL_Name` incluye los siguientes parámetros:

pool_name

Un nombre de agrupación de hasta 13 caracteres. El nombre de agrupación puede incluir cualquier carácter a excepción del signo de porcentaje ("%"), el guion bajo ("_") y el asterisco ("*").

POOLType

El tipo de agrupación: *Scratch* (el valor por defecto) o *Mvc*.

POOLId (nnnnn)

De manera opcional, especifica el ID de la agrupación (*nnnnn*).

- Si decide especificar este parámetro, debe indicar un ID de agrupación que no exista actualmente.
- Si no se especifica este parámetro, el sistema asigna automáticamente un ID de agrupación igual al primer valor sin usar a partir de 10000.

Ejemplo

En el siguiente ejemplo, el usuario ejecuta el comando `DEFINE POOL_Name` para definir la agrupación reutilizable `SCRPOOL1`:

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

DEFINE POOL_Access

En la siguiente sección, se describe el comando `DEFINE POOL_Access`.

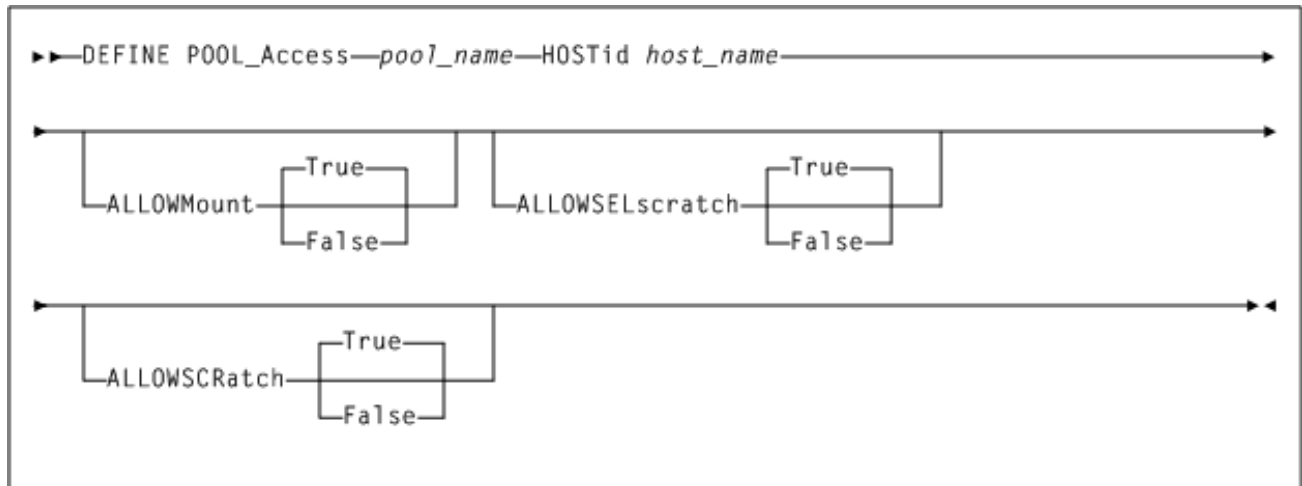
Descripción

El comando `DEFINE POOL_Access` define el acceso permitido para un *pool_name* mediante un host especificado. Si no se define ningún acceso de agrupación para un *pool_name* específico, se permite el acceso total.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando `DEFINE POOL_Access`:

Figura 3.2. Sintaxis del comando DEFINE POOL_Access



Parámetros

Como se muestra en [Figura 3.2, “Sintaxis del comando DEFINE POOL_Access”](#), el comando `DEFINE POOL_Access` incluye los siguientes parámetros:

pool_name

Un nombre de agrupación de hasta 13 caracteres. El nombre de agrupación puede incluir cualquier carácter a excepción del signo de porcentaje ("%"), el guion bajo ("_") y el asterisco ("*"). El *pool_name* especificado debe haberse definido previamente utilizando el comando `DEFINE POOL_Name` con el tipo de agrupación `SCRATCH` (no se define el acceso para las agrupaciones MVC). Consulte "[DEFINE POOL_Name](#)" para obtener más información.

HOSTid *host_name*

Especifica un nombre de host de hasta 31 caracteres. Este parámetro es obligatorio.

El *HOSTid* especificado es la parte del nombre de dominio completo de Internet que hace referencia al nombre de host. Por ejemplo, si el nombre de dominio completo es *myhost.plus.domain.name*, el *HOSTid* especificado es *myhost*.

ALLOWMount

De manera opcional, especifica el acceso de montaje. Puede especificar *True* (Verdadero) o *False* (Falso). Este parámetro solo es válido para una agrupación definida como `POOLTYPE MVC`.

ALLOWSELscratch

De manera opcional, especifica el acceso para los comandos de XAPI *query_scr_mnt_info*, *sel_scratch* y *mount* para un volumen reutilizable. Puede especificar

True (Verdadero) o *False* (Falso). Si no especifica este parámetro, se establece en *True* (Verdadero) por defecto.

Este parámetro solo es válido para una agrupación definida como *POOLTYPE SCRATCH*. Si se especifica este parámetro para una agrupación definida como *MVC*, el parámetro recibe automáticamente el valor por defecto *False* (Falso).

ALLOWSCRatch

De manera opcional, especifica el acceso reutilizable. Puede especificar *True* (Verdadero) o *False* (Falso). Si no especifica este parámetro, se establece en *True* (Verdadero) por defecto.

Este parámetro solo es válido para una agrupación definida como *POOLTYPE SCRATCH*. Si se especifica este parámetro para una agrupación definida como *MVC*, el parámetro recibe automáticamente el valor por defecto *False* (Falso).

Ejemplo

En el siguiente ejemplo, el usuario ejecuta el comando *DEFINE POOL_Access* para definir el acceso para el nombre de agrupación reutilizable *SCRPOOL1*, donde el host es *PRODMVS*, y *ALLOWMOUNT* y *ALLOWSCRATCH* son *True* (Verdadero):

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

DELETE POOL_Name

En la siguiente sección, se describe el comando *DELETE POOL_Name*.

Descripción

El comando *DELETE POOL_Name* suprime una agrupación. Este comando es similar al comando *delete pool* de ACSLS, con las siguientes excepciones:

- La entrada especifica un *nombre de agrupación* en lugar de un *ID de agrupación*.
- El usuario puede elegir que los volúmenes asignados a la agrupación se reasignen a la agrupación 0.

Consulte la *Guía del administrador de StorageTek Automated Cartridge System Library Software (ACSL)* para obtener más información sobre el comando *delete pool* de ACSLS.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando *DELETE POOL_Name*:

Figura 3.3. Sintaxis del comando DELETE POOL_Name

Parámetros

Como se muestra en la [Figura 3.3, “Sintaxis del comando DELETE POOL_Name”](#), el comando `DELETE POOL_Name` incluye los siguientes parámetros:

pool_name

Un nombre de agrupación de hasta 13 caracteres. El nombre de agrupación puede incluir cualquier carácter a excepción del signo de porcentaje ("%"), el guion bajo ("_") y el asterisco (*). El *pool_name* especificado debe haberse definido previamente utilizando el comando `DEFINE POOL_Name`. Consulte "[DEFINE POOL_Name](#)" para obtener más información.

RESETvols

Indica que los volúmenes actualmente asignados a la agrupación que se suprime deben reasignarse a la agrupación 0. Si no se especifica este parámetro y existen volúmenes actualmente dentro de esta agrupación, el comando `delete pool-name` genera un error.

Ejemplo

En el siguiente ejemplo, el usuario ejecuta el comando `DELETE POOL_Name` para suprimir la agrupación `SCRPOOL1` y reasignar sus volúmenes a la agrupación 0 (la agrupación por defecto):

```
XCMD DELETE POOL_NAME SCRPOOL_TEST1 RESETVOLS
```

DELETE POOL_Access

En la siguiente sección, se describe el comando `DELETE POOL_Access`.

Descripción

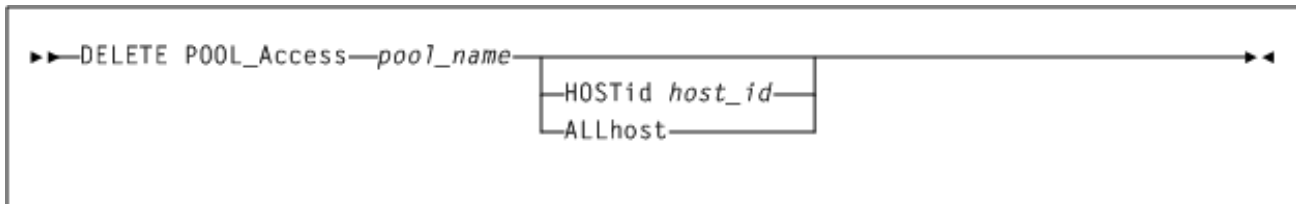
El comando `DELETE POOL_Access` suprime las reglas de acceso a agrupación asociadas con un solo nombre de host (*host_id*) o suprime todas las reglas de acceso a agrupación de todos los hosts.

- Si se especifica un nombre de host (*host_id*), se suprimen solo las reglas de acceso del host especificado en la agrupación indicada.
- Si se especifica `ALLhost`, se suprimen todas las reglas de acceso a la agrupación.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando *DELETE POOL_Access*:

Figura 3.4. sintaxis del comando DELETE POOL_Access



Parámetros

Como se muestra en [Figura 3.4, “sintaxis del comando DELETE POOL_Access”](#), el comando *DELETE POOL_Access* incluye los siguientes parámetros:

pool_name

Un nombre de agrupación de hasta 13 caracteres. El nombre de agrupación puede incluir cualquier carácter a excepción del signo de porcentaje ("%"), el guion bajo ("_") y el asterisco ("*"). El *pool_name* especificado debe haberse definido previamente utilizando el comando *DEFINE POOL_Name*. Consulte "[DEFINE POOL_Name](#)" para obtener más información.

HOSTid host_name

De manera opcional, especifica un nombre de host de hasta 31 caracteres e indica que se deben eliminar las restricciones de acceso para este ID de host (sin cambios en las reglas de acceso del resto de los ID de host). Este parámetro y *ALLhost* se excluyen mutuamente.

ALLhost

Opcionalmente, especifica que se eliminen todas las restricciones de acceso para el *pool_name* específico de todos los ID de host. Este parámetro y *HOSTid* se excluyen mutuamente.

Ejemplo

En el siguiente ejemplo, el usuario ejecuta el comando *DELETE POOL_Access* para suprimir el acceso a la agrupación del host *PRODMVS* a *SCRPOOL1*:

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

SET POOL_Name

En la siguiente sección, se describe el comando *SET POOL_Name*.

Descripción

El comando `SET POOL_Name` asigna volúmenes a un nombre de agrupación o establece el límite inferior de un nombre de agrupación.

Nota:

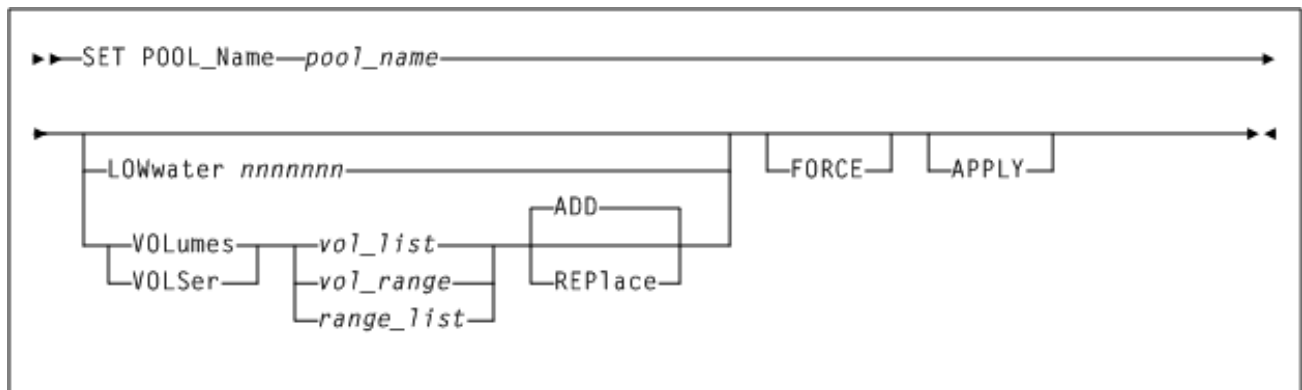
Si la sintaxis del comando es válida, se agregan los rangos de volumen especificados a la base de datos aunque algunos volúmenes dentro del rango no se puedan agregar a la agrupación. Los volúmenes no se pueden agregar a la agrupación si alguna de las siguientes condiciones es verdadera:

- El volumen tiene el valor `LOCKed` (Bloqueado).
- El volumen ya está en una agrupación diferente, a menos que se especifique `FORCE`.
- El volumen está en una biblioteca lógica.
- El volumen tiene un propietario de ACSLS y el control de acceso está en vigor.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando `SET POOL_Name`:

Figura 3.5. Sintaxis del comando SET POOL_Name



Parámetros

Como se muestra en la [Figura 3.5, “Sintaxis del comando SET POOL_Name”](#), el comando `SET POOL_Name` incluye los siguientes parámetros:

pool_name

Un nombre de agrupación de hasta 13 caracteres. El nombre de agrupación puede incluir cualquier carácter a excepción del signo de porcentaje ("%"), el guion bajo ("_") y el asterisco ("*"). El *pool_name* especificado debe haberse definido previamente utilizando el comando `DEFINE POOL_Name`. Consulte "[DELETE POOL_Name](#)" para obtener más información.

LOWwater nnnnnnn

De manera opcional, define el número mínimo de volúmenes disponibles que debe contener la agrupación. Si el número de volúmenes disponibles cae por debajo del límite inferior, el sistema ACSLS genera una advertencia.

VOLumes o VOLSer (*volser-list, volser-range, range-list*)

De manera opcional, especifica una lista de volsers, un rango de volsers o una lista de rangos de volsers. Puede especificar un máximo de 16 volúmenes simples o rangos.

Los rangos de volúmenes se evalúan según los rangos de volsers con formato ACSLS (secuencia de ordenación ACSII), y no es necesario que cumplan con los rangos de patrones de volumen HSC; sin embargo, ambos números de serie de volumen en un solo rango deben contener el mismo número de caracteres, y solo se consideran dentro del rango los números de serie de volumen con la misma cantidad de caracteres. Por ejemplo, el rango 1234-12356 no es válido. En el rango 1234-2345, un número de serie 123567 no se considera dentro del rango. Si los rangos especificados son válidos, se agregan a la base de datos aunque algunos volúmenes dentro del rango no se puedan asignar a la agrupación. Los rangos de volúmenes especificados no deben superponerse entre ellos ni con ninguno de los rangos de volúmenes existentes.

Los rangos de volúmenes se evalúan según los rangos de volsers con formato ACSLS:

- Ambos volsers especificados en el rango deben contener el mismo número de caracteres.
- El primer elemento de volser en el rango debe ser estrictamente menor que el segundo elemento de volser.
- Si los volsers especificados tienen menos de 6 caracteres de longitud, los volsers se evalúan como si fueran campos de 6 caracteres en blanco y alineados a la izquierda.
- En el rango solo hay volsers estrictamente mayores o iguales que el primer elemento de volser y estrictamente menores o iguales que el segundo elemento de volser cuando se evalúan utilizando la secuencia de ordenación ASCII. Por ejemplo, el rango de volsers 123456-234567 contiene el volser 12345A.

Si los rangos especificados son válidos, se agregan a la base de datos de *pool_name* aunque algunos volúmenes dentro del rango no se puedan asignar a la agrupación. Los rangos de volúmenes especificados no deben superponerse entre ellos ni con ninguno de los rangos de volúmenes existentes.

ADD

Indica que los rangos de volúmenes especificados se agregarán a los rangos existentes en la agrupación. Este es el valor por defecto. Si se especifica, también se debe especificar VOLumes. *ADD* y *REPLace* se excluyen mutuamente.

REPlace

Indica que los rangos de volúmenes especificados deben sustituir los rangos de volúmenes especificados actualmente y que los volúmenes en los rangos anteriores se deben reasignar a la agrupación 0. Si se especifica, también se debe especificar *VOLumes* (o *VOLSer*). *REPLace* y *ADD* se excluyen mutuamente.

FORCE

De manera opcional, indica que los volúmenes ya asignados a otras agrupaciones se deben reasignar a la agrupación nueva. Si no se especifica este parámetro, no se actualiza ningún volumen actualmente asignado a una agrupación diferente. Si se especifica, también se debe especificar *VOLumes* (o *VOLSer*).

APPLY

De manera opcional, indica que los rangos de volúmenes definidos anteriormente se deben aplicar a los volúmenes de la base de datos. Este parámetro se puede usar cuando no se hayan actualizado algunos volúmenes debido a las restricciones que se detallaron anteriormente. Si se especifica, también se debe especificar *VOLumes* (o *VOLSer*).

Ejemplo

En el siguiente ejemplo, el comando agrega los volúmenes *DRL001-DRL999* a *SCRPOOL1*:

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

QUERY POOL_Name

En la siguiente sección, se describe el comando *QUERY POOL_Name*.

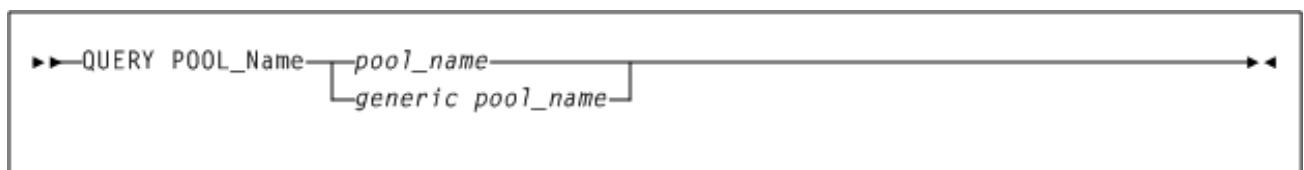
Descripción

El comando *QUERY POOL_Name* muestra todas las agrupaciones, las agrupaciones que comienzan con los caracteres especificados o una sola agrupación.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando *QUERY POOL_Name*:

Figura 3.6. Sintaxis del comando QUERY POOL_Name

**Parámetros**

Como se muestra en la [Figura 3.6, “Sintaxis del comando QUERY POOL_Name”](#), el comando *QUERY POOL_Name* incluye los siguientes parámetros:

pool_name

Un nombre de agrupación existente de hasta 13 caracteres o un nombre genérico con un carácter de asterisco (“*”) al final. Este parámetro es obligatorio.

El formato *pool_name* determina la salida que se muestra de la siguiente manera:

- Cuando *pool_name* no es un nombre genérico, la información de agrupación que se muestra incluye los volsers y los rangos de volsers definidos para la única agrupación coincidente.
- Cuando *pool_name* es un nombre genérico (incluye un "*" final), la información de agrupación que se muestra no incluye los volsers y los rangos de volsers definidos para las agrupaciones coincidentes.

Ejemplo

En el siguiente ejemplo, el usuario ejecuta el comando *QUERY POOL_Name* para mostrar información sobre pool1, incluidos los rangos de volúmenes asociados.

```
XCMD QUERY POOL_NAME POOL1
```

En el siguiente ejemplo, el usuario ejecuta el comando *QUERY POOL_Name* para mostrar información sobre todas las agrupaciones mencionadas (sin incluir los rangos de volúmenes):

```
XCMD QUERY POOL_NAME *
```

En el siguiente ejemplo, el usuario ejecuta el comando *QUERY POOL_Name* para mostrar información sobre todas las agrupaciones que comienzan con "A" (sin incluir los rangos de volúmenes):

```
XCMD QUERY POOL_NAME A*
```

QUERY POOL_Access

En la siguiente sección, se describe el comando *QUERY POOL_Access*.

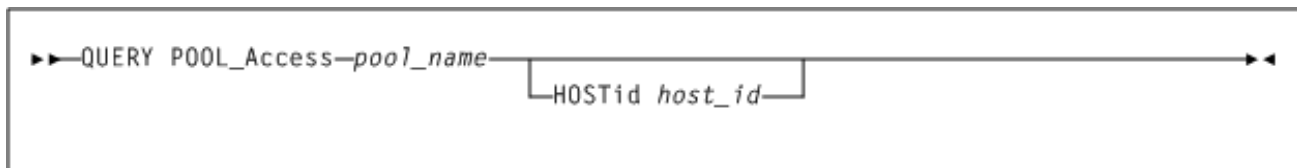
Descripción

El comando *QUERY POOL_Access* muestra las reglas de acceso a agrupación de un solo host o de todos los hosts asociados con un nombre de agrupación.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando *QUERY POOL_Access*:

Figura 3.7. Sintaxis del comando QUERY POOL_Access



Parámetros

Como se muestra en la [Figura 3.7, “Sintaxis del comando QUERY POOL_Access”](#), el comando `QUERY POOL_Access` incluye los siguientes parámetros:

pool_name

Un nombre de agrupación de hasta 13 caracteres. El nombre de agrupación puede incluir cualquier carácter a excepción del signo de porcentaje ("%"), el guion bajo ("_") y el asterisco ("*"). El *pool_name* especificado debe haberse definido previamente utilizando el comando `DEFINE POOL_Name`. Consulte "[DEFINE POOL_Name](#)" para obtener más información.

HOSTid *host_id*

De manera opcional, especifica un nombre de host de 1 a 31 caracteres. El *host_id* especificado debe haberse definido previamente para el *pool_name* utilizando el comando `DEFINE POOL_Access`. Consulte "[DEFINE POOL_Access](#)" para obtener más información.

Si no se especifica este parámetro, se muestran las reglas de acceso de todos los hosts asociados con la agrupación.

Ejemplo

En el siguiente ejemplo, el usuario ejecuta el comando `QUERY POOL_Access` para consultar las reglas de acceso de todos los hosts asociados con el nombre de agrupación `SCRPOOL1`:

```
XCMD QUERY POOL_ACCESS SCRPOOL1
```

Capítulo 4. Comandos del operador del servidor de la XAPI de XCMD

En este capítulo, se describen los comandos del operador del servidor de la XAPI admitidos por XCMD. Estos comandos le permiten a un operador o administrador supervisar el componente XAPI de ACSLS. Los comandos admitidos incluyen:

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

Ejecute estos comandos desde `user_proc` de ACSLS utilizando el comando `XCMD`. Consulte [Capítulo 2, Convenciones de comandos XCMD](#) para obtener más información.

Consulte "[Convenciones de sintaxis](#)" para obtener información sobre las convenciones utilizadas en los diagramas de flujo de sintaxis incluidos en las secciones de comandos a continuación.

LOG

En la siguiente sección, se describe el comando `LOG`.

Descripción

El comando `LOG` cambia o muestra la configuración de log actual del servidor de la XAPI.

Nota:

Las variables de entorno de ACSLS establecen el nombre y la ubicación del archivo log, cuyos valores por defecto son el nombre de archivo `vlog.file` y el directorio `XAPI_WORK_PATH`.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando `LOG`:

Figura 4.1. Sintaxis del comando LOG



Parámetros

Como se muestra en la [Figura 4.1, “Sintaxis del comando LOG”](#), el comando *LOG* incluye los siguientes parámetros:

LIst

De manera opcional, especifica que se muestre la configuración de log actual. Si no se especifican parámetros de comando, el valor por defecto es *LIst*.

log_setting

De manera opcional, indica la configuración de log deseada como una serie de caracteres "0" y "1".

- *0* significa que la configuración de log posicional deseada debe ser *OFF* o desactivado.
- *1* significa que la configuración de log posicional deseada debe ser *ON* o activado.

El valor especificado sustituye completamente la configuración actual de log y no se fusiona con esta. Las opciones de configuración de log posicional son las siguientes:

- *1*: registrar los mensajes de error en el log del componente de XAPI de ACSLS.
- *01*: registrar los mensajes en el archivo log.
- *001*: registrar los errores de transacción de entradas en el archivo log.
- *0001*: registrar todos los paquetes XML *recv()* en el archivo log.
- *00001*: registrar todos los paquetes XML *send()* en el archivo log.
- *000001*: registrar los comandos locales y las respuestas en el archivo log.

Los mensajes de *LOG* de XAPI se guardarán en el archivo *\$ACS_HOME/log/xapi/vlog.file*.

Nota:

El comando *LOG* le permite establecer y mostrar más posiciones de ceros y unos (16) que las definidas actualmente como las configuraciones de *LOG* anteriores. Esto permite una expansión futura, y las configuraciones extrañas de ceros y unos de *LOG* simplemente se ignoran.

OFF

De manera opcional, especifica que se debe desactivar el registro. Esto es equivalente a *LOG 0*.

Ejemplo

En el siguiente ejemplo, un usuario ejecuta el comando *LOG* para configurar el servidor de la XAPI para que registre todos los errores y todos los paquetes de XML recv() en el archivo log:

```
XCMD LOG 1011
```

LIST

En la siguiente sección, se describe el comando *LIST*.

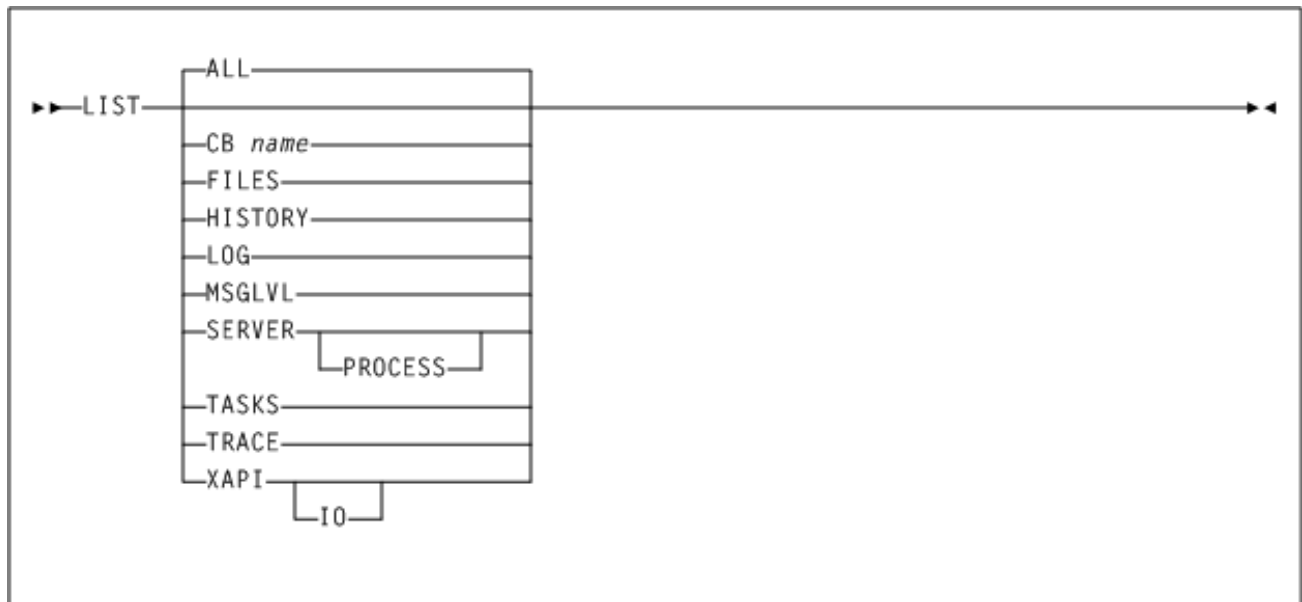
Descripción

El comando *LIST* muestra la configuración y el estado actuales del servidor de la XAPI.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando *LIST*:

Figura 4.2. Sintaxis del comando LIST



Parámetros

Como se muestra en la [Figura 4.2, “Sintaxis del comando LIST”](#), el comando *LIST* incluye los siguientes parámetros:

ALL

De manera opcional, muestra todos los estados y la configuración del servidor de la XAPI. Este es el parámetro por defecto si se ejecuta el comando *LIST* sin parámetros.

CB name

Especifica que el bloque de control mencionado se muestra en formato de caracteres y caracteres hexadecimales. Úselo solo cuando el soporte de Oracle se lo indique.

Especifique uno de los siguientes bloques de control para *name*:

- *HTTPCVT*
- *HTTGPL*
- *HTTPREQ-*nnn**
- *HTTPAPI-*nnn**

Para *LIST CB HTTPREQ* y *LIST CB HTTPAPI*, se debe especificar un índice entre 0 y 999. Además, el índice debe estar delimitado respecto del nombre del bloque de control mediante un guion simple ("-") sin espacios. Por ejemplo:

```
LIST HTTPREQ-0
```

0

```
LIST HTTPAPI-900
```

FILES

De manera opcional, especifica que se muestre el nombre de ruta completo de los archivos log, de control y de rastreo del servidor de la XAPI.

HISTORY

De manera opcional, especifica que se muestre el historial de recuento de transacciones de XAPI para las últimas 24 horas.

LOG

De manera opcional, especifica que se muestre la configuración de log actual del servidor de la XAPI. Esto es equivalente a ejecutar un comando *LOG* sin parámetros.

MSGLVL

De manera opcional, especifica que se muestre la configuración de nivel de mensaje actual del servidor de la XAPI. Esto es equivalente a ejecutar un comando *MSGLVL* sin parámetros.

SERVER

De manera opcional, especifica que se muestren las versiones actuales de UNIX y del servidor de la XAPI, los parámetros relevantes y las variables de entorno, además de las colas de mensajes y los segmentos compartidos.

También puede incluir la palabra clave *PROCESS* para solicitar que se muestren todos los ID de los procesos activos del servidor de la XAPI, los recuentos de archivos abiertos y threads, y el uso de memoria y CPU.

PROCESS

Cuando se especifica con *SERVER*, la palabra clave *PROCESS* solicita que también se muestren todos los ID de los procesos activos del servidor de la XAPI, los recuentos de archivos abiertos y threads, y el uso de memoria y CPU.

TASKS

De manera opcional, especifica que se muestren las tareas actuales de trabajo y del sistema del servidor de la XAPI.

TRACE

De manera opcional, especifica que se muestre la configuración de rastreo actual del servidor de la XAPI. Esto es equivalente a ejecutar un comando *TRACE* sin parámetros.

XAPI

De manera opcional, especifica que se muestren el puerto del listener y la dirección IP actuales del servidor de la XAPI.

Además, puede incluir la palabra clave *IO* para solicitar que también se muestren todas las estadísticas del listener del servidor de la XAPI.

Ejemplo

En el siguiente ejemplo, un usuario ejecuta el comando *List* para mostrar el puerto del listener, la dirección IP y el estado de listener actuales del servidor de la XAPI:

```
XCMD LIST XAPI IO
```

MSGLVL

En la siguiente sección, se describe el comando *MSGLVL*.

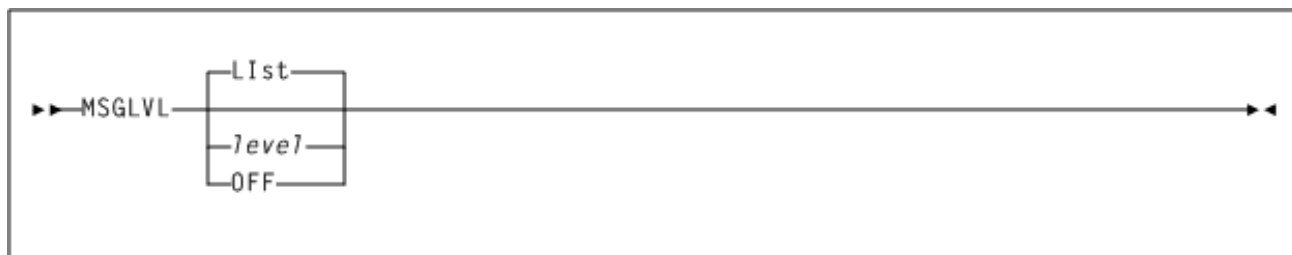
Descripción

El comando *MSGLVL* cambia o muestra el nivel de detalle del mensaje del servidor de la XAPI.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando *MSGLVL*:

Figura 4.3. Sintaxis del comando MSGLVL



Parámetros

Como se muestra en [Figura 4.3, “Sintaxis del comando MSGLVL”](#), el comando *MSGLVL* incluye los siguientes parámetros:

L*Ist*

Especifica que se muestre la configuración de *MSGLVL* actual. Si no se especifican parámetros de comando, el valor por defecto es *L**Ist*.

level

Especifica la configuración de *MSGLVL* deseada. Cuanto mayor sea el nivel, más detallados serán los mensajes del servidor de la XAPI. Los niveles de mensaje son acumulativos; *MSGLVL 8* genera todos los mensajes hasta *MSGLVL 8* inclusive (es decir, genera los mensajes de *MSGLVL 0* a *8*). Los niveles de mensaje se generalizan de la siguiente manera:

- 0: mensajes de inicio, cierre y error normales; estos mensajes no se pueden suprimir.
- 4: advertencias graves.
- 8: advertencias menores.
- 12: mensajes de opciones y parámetros de inicio.
- 16: mensajes adicionales de inicio y cierre.
- 20: mensajes de nivel de diagnóstico 20 adicionales.
- 24: mensajes de nivel de diagnóstico 24 adicionales.
- 28: mensajes adicionales de inicio y cierre de tarea.

OFF

Especifica que se deben desactivar todos los mensajes detallados. Esto es equivalente a *MSGLVL 0*.

Ejemplo

En el siguiente ejemplo, un usuario ejecuta el comando *MSGLVL* para cambiar los mensajes del servidor de la XAPI para incluir todas las advertencias:

```
XCMD MSGVLV 8
```

TRACE

En la siguiente sección, se describe el comando *TRACE*.

Descripción

El comando *TRACE* cambia o muestra la configuración de rastreo del servidor de la XAPI.

Nota:

- Las variables de entorno de ACSLS establecen el nombre y la ubicación del archivo de rastreo, cuyos valores por defecto son el nombre de archivo *vtrace.file* y la variable *DV_TAG_XAPI_WORK_PATH*.
- El rastreo puede afectar de manera significativa al rendimiento del sistema. Solo active el rastreo si se lo solicita el soporte de Oracle StorageTek.
- El comando *TRACE* le permite establecer y mostrar más posiciones de ceros y unos (16) que las definidas actualmente como las configuraciones de *TRACE* anteriores. Esto permite una expansión futura, y las configuraciones extrañas de ceros y unos de *TRACE* simplemente se ignoran.

Sintaxis

En la siguiente figura, se muestra la sintaxis del comando *TRACE*:

Figura 4.4. Sintaxis del comando TRACE



Parámetros

Como se muestra en la [Figura 4.4, “Sintaxis del comando TRACE”](#), el comando *TRACE* incluye los siguientes parámetros:

List

De manera opcional, especifica que se muestre la configuración de rastreo actual. Si no se especifican parámetros de comando, el valor por defecto es *List*.

trace_setting

De manera opcional, indica la configuración de rastreo deseada como una serie de caracteres "0" y "1". 0 significa que la configuración de rastreo posicional deseada debe ser *OFF* o desactivado, mientras que 1 significa que la configuración de rastreo posicional deseada debe ser *ON* o activado. El valor especificado sustituye completamente la configuración actual de rastreo y no se fusiona con esta. Las opciones de configuración de rastreo posicional son las siguientes:

- *1*: rastrea errores y los registra en el archivo de rastreo.
- *01*: rastrea eventos y funciones TCP/IP y los registra en el archivo de rastreo.
- *001*: rastrea eventos y funciones PGMI o ACSAPI y los registra en el archivo de rastreo.
- *0001*: rastrea eventos del servidor de la XAPI no clasificados de otro modo y los registra en el archivo de rastreo.

- *00001*: rastrea eventos `malloc()` y `free()` y los registra en el archivo de rastreo.
- *000001*: rastrea eventos del analizador XML y los registra en el archivo de rastreo.
- *0000001*: rastrea eventos de procesos del servidor de comandos y los registra en el archivo de rastreo.
- *00000001*: rastrea eventos del proceso de supervisión y los registra en el archivo de rastreo.
- *000000001*: rastrea eventos y funciones CSV y los registra en el archivo de rastreo.

OFF

De manera opcional, especifica que el rastreo está desactivado. Esto es equivalente a *TRACE 0*.

Ejemplo

En el siguiente ejemplo, un usuario ejecuta el comando *TRACE* para configurar el servidor de la XAPI con el fin de que rastree todos los errores y todos los eventos `malloc()` y `free()` y los registre en el archivo de rastreo:

```
XCMD TRACE 10001
```


Capítulo 5. Compatibilidad de XCMD con solicitudes de usuarios de XAPI y comandos de HSC/VTCS

Además de los comandos de administración y configuración del servidor de la XAPI descritos en el [Capítulo 3, Comandos de configuración y administración de XAPI de XCMD](#) y los comandos del operador del servidor de la XAPI descritos en el [Capítulo 4, Comandos del operador del servidor de la XAPI de XCMD](#), la interfaz XCMD también admite muchas solicitudes de usuario de XAPI y es compatible con comandos de texto de HSC/MVS.

Por ejemplo, para introducir una solicitud `QUERY_VOLUME` desde `user_proc` de ACSLS con el comando XCMD:

```
XCMD QUERY_VOLSER S00001
```

Solicitudes, comandos y parámetros de XAPI admitidos

En la siguiente tabla, se muestran las solicitudes de XAPI y los comandos de texto de HSC/MVS correspondientes admitidos por XCMD:

Tabla 5.1. Solicitudes de XAPI y comandos de HSC/MVS correspondientes admitidos por XCMD

Solicitud de XAPI	Comando de texto de HSC/MVS correspondiente
<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>	No tiene ningún comando de texto de HSC/MVS correspondiente
<code>QUERY_SCRATCH</code>	<code>Display SCRATCH</code>

Solicitud de XAPI	Comando de texto de HSC/MVS correspondiente
<i>QUERY_THRESHOLD</i>	<i>Display THReshold</i>
<i>QUERY_VOLSER</i> o <i>QUERY_VOLUME</i>	<i>Display Volser</i>
<i>QUERY_VOLUME_INFO</i>	<i>Display VOLume_info</i>
<i>SCRATCH</i>	<i>SCRAtch</i>
<i>UNSCRATCH</i>	<i>UNSCRatch</i>
<i>VOLRPT</i> ¹	<i>VOLRpt</i>

¹Esta solicitud *VOLRPT* es el comando y el informe *VOLRPT* con formato HSC, NO el *volrpt* con formato ACSLS.

Consulte la *Referencia de comandos, sentencias de control y utilidades de ELS* para obtener más información sobre los comandos de texto de HSC/MVS que se muestran arriba.

Aunque se admiten estas solicitudes y comandos, no todos los parámetros son válidos en ACSLS. Los parámetros no válidos son los siguientes:

- *DISMOUNT* (o el comando *DISMount*)

Si se especifica lo siguiente, se produce un error de sintaxis:

- Dirección de dispositivo de MVS en formato ccuu
- Parámetro posicional *hostid*

- *EJECT* (o el comando *EJect*)

La especificación de los siguientes parámetros produce un error de sintaxis:

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

- *MOUNT* (o el comando *Mount*)

Si se especifica lo siguiente, se produce un error de sintaxis:

- Dirección de dispositivo de MVS en formato ccuu
- Parámetro posicional *hostid*
- El parámetro *MGMTCLAS* no es válido

- *MOVE* (o el comando *MOVE*)

Se admiten todos los parámetros.

- *QUERY_CAP* (o el comando *Display Cap*)
 - Se admiten todos los parámetros.
- *QUERY_DRIVES* (o el comando *Display DRives*)

La especificación de los siguientes parámetros produce un error de sintaxis:

- *ALL*
- *IDLE*

- *LIBRARY*
- *DETAIL*
- *IDENTITY*
- *MEDIA*
- *RECTECH*
- *UNIT*
- *BYDRIVE*
- *BYLOC*
- *SHOWLSLOT*
- *QUERY_DRIVE_INFO* (o el comando *Display DRIVE_INFO*)

La especificación de los siguientes parámetros produce un error de sintaxis:

- *ALL*
- *IDLE*
- *LIBRARY*
- *DETAIL*
- *IDENTITY*
- *MEDIA*
- *RECTECH*
- *VIRTUAL*
- *DEVADDR*
- *LSMLOC*
- *QUERY_LSM* (o el comando *Display Lsm*)
Se admiten todos los parámetros.
- *QUERY_SERVER* (o el comando *Display SERVER*)
Se admiten todos los parámetros.
- *QUERY_SCRATCH* (o el comando *Display SCRatch*)
Se admiten todos los parámetros.
- *QUERY_THRESHOLD* (o el comando *Display THReshold*)
Se admiten todos los parámetros.
- *QUERY_VOLSER* (o el comando *Display Volser*)
Se admiten todos los parámetros.
- *QUERY_VOLUME_INFO* (o el comando *Display VOLume_info*)
Se admiten todos los parámetros.
- *SCRATCH* (o el comando *SCRAtch*)

Se admiten todos los parámetros.

- *UNSCRATCH* (o el comando *UNSCRatch*)

Se admiten todos los parámetros.

- *VOLRPT* (o el comando *VOLRpt*)

Si se especifica lo siguiente, se produce un error de sintaxis:

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

Ejecución de comandos para el servidor de la XAPI de ACSLS utilizando el comando route del cliente de VM o SMC

A excepción de los comandos *EJECT* y *VOLRpt*, puede ejecutar cualquiera de los comandos de HSC/MVS que se muestran en la [Tabla 5.1, “Solicitudes de XAPI y comandos de HSC/MVS correspondientes admitidos por XCMD”](#) desde un cliente de SMC/MVS o de VM para el servidor de la XAPI de ACSLS utilizando el comando *Route*.

A diferencia de los comandos *XCMD* introducidos desde *cmd_proc* de ACSLS, el comando *Route* de cliente de VM o SMC/MVS cumple con las convenciones de sintaxis establecidas de SMC/HSC. Además, se omite el token *XCMD*.

Por ejemplo:

Para ejecutar una solicitud *QUERY VOLUME* de XAPI como comando *XCMD* de *cmd_proc* de ACSLS:

```
XCMD QUERY VOLUME VOLSER
```

Donde *VOLSER* es el volumen deseado.

Para ejecutar la solicitud *QUERY VOLUME* de la XAPI desde un cliente SMC/MVS para el servidor de la XAPI de ACSLS utilizando el comando *ROUTE*:

```
ROUTE ACSLS, QUERY VOLUME(vo1ser)
```

Donde *vo1ser* es el volumen deseado.

Consulte la *Referencia de comandos, sentencias de control y utilidades de ELS* para obtener más información sobre el comando *ROUTE* del SMC.

Capítulo 6. Variables de control del servidor de la XAPI de ACSLS

El software ACSLS de Oracle proporciona un juego de variables de sistema que permite controlar ciertos comportamientos del sistema ACSLS. Se incluyen varias variables de control específicas para el componente de servidor de la XAPI de ACSLS. Entre estas variables, se incluyen las siguientes:

- [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](#)

Configuración de variables de control del servidor de la XAPI de ACSLS

Use la utilidad *acsss_config* o *dv_config* de ACSLS para visualizar y configurar las variables estáticas de la XAPI de ACSLS. Debe reiniciar ACSLS para que se apliquen los cambios.

- Para usar la utilidad *dv_config*, introduzca uno de los siguientes comandos:
 - *dv_config -d* para mostrar todas las variables de ACSLS.
 - *dv_config -p <variable_name>* para actualizar la variable de XAPI.
- Para usar la utilidad *acsss_config*, use la secuencia de comandos *acsss_config* de ACSLS para acceder a la pantalla ACSLS Feature Configuration (Configuración de la función ACSLS):

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

Seleccione la opción 9 *Set XAPI server variables* (Configurar variables del servidor de la XAPI).

Nota:

Esta opción está visible solo si tiene activada la función de servidor de la XAPI de ACSLS.

Consulte la *Guía del administrador de StorageTek Automated Cartridge System Library Software* para obtener más información sobre las utilidades *acsss_config* y *dv_config* y sobre cómo usarlas para mostrar y actualizar las variables de ACSLS.

Descripciones de las variables de XAPI de ACSLS

En esta sección, se describen las variables válidas utilizadas para controlar el componente de servidor de la XAPI de ACSLS.

XAPI_PORT

Petición de datos: *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]* (Los cambios en el puerto de entrada definido por el usuario hacia el servidor de la XAPI no tienen efecto hasta que se reinicie el servidor de la XAPI. Número de puerto utilizado por el servidor de la XAPI para recibir solicitudes de entrada de XAPI. [50020]):

Esta opción especifica el puerto que usa el servidor de la XAPI para las solicitudes TCP entrantes provenientes de clientes. Introduzca un número entre 1024 y 65535 para definir el puerto que usa el servidor de la XAPI. No especifique el puerto 50003. Para obtener más información, consulte la *Guía de seguridad de ACSLS*.

Se debe reiniciar el servidor de la XAPI para que se aplique esta variable.

XAPI_WORK_PATH

Petición de datos: *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]* (Los cambios en el directorio de trabajo de la XAPI no tienen efecto hasta que se reinicie el servidor de la XAPI. Coloque la información de rastreo y de log de XAPI en el directorio. [\$ACS_HOME/log/xapi]):

Seleccione un directorio en donde se colocarán los archivos de trabajo del servidor de la XAPI. Por defecto, el servidor de la XAPI registra la información en el directorio `$ACS_HOME/log/xapi`. Con el uso normal, el valor de esta variable no se modifica. Se puede especificar una ruta alternativa si hay problemas de espacio de disco en el sistema de archivos que contiene `$ACS_HOME`. La ruta asignada debe ser una ruta absoluta (por ejemplo, debe comenzar con `/` o `$ACS_HOME`). Se debe reiniciar el servidor de la XAPI para que se aplique esta variable.

XAPI_LOG_SIZE

Petición de datos: *Changes to the XAPI log size will not take effect until the XAPI server is restarted. Maximum XAPI log size in Mbytes [20]* (Los cambios en el tamaño de log de la XAPI no se aplicarán hasta que el servidor de la XAPI se reinicie. Tamaño máximo de log de la XAPI en megabytes. [20]):

Esta opción especifica el tamaño máximo para el log de la XAPI, expresado en megabytes (definido aquí como "1048576 bytes."). Introduzca un número no negativo. El valor por defecto de esta opción es 20.

Se debe reiniciar el servidor de la XAPI para que se aplique esta variable.

XAPI_LOG_FILE_NUM

Petición de datos: *Number of XAPI Log archive files to retain [10]* (Número de archivos de almacenamiento de log de la XAPI que se conservan [10]):

Esta opción especifica el número de archivos log de la XAPI almacenados que se deben conservar.

Cuando el tamaño de `vlog.file` excede el tamaño máximo, se cambia el sufijo del nombre del archivo log de 0 a *n*. El archivo más reciente es el que tiene el 0; el que tiene *n* es el más antiguo. Los archivos almacenados se guardan en el directorio `XAPI_WORK_PATH`.

Cuando se alcanza el número especificado de logs almacenados, el archivo más antiguo se elimina del directorio de almacenamiento cada vez que se agrega uno nuevo a ese directorio. Puede conservar entre uno y 99 archivos almacenados.

Introduzca un número entre 1 y 99 para especificar la cantidad de archivos log almacenados que desea conservar.

Se debe reiniciar el servidor de la XAPI para que se aplique esta variable.

XAPI_TRACE_SIZE

Petición de datos: *Changes to the XAPI trace size will not take effect until the xapi server is restarted. Maximum XAPI trace size in Mbytes. [50]* (Los cambios en el tamaño de rastreo de la XAPI no se aplicarán hasta que el servidor de la XAPI se reinicie. Tamaño máximo de rastreo de la XAPI en megabytes. [50]):

Esta opción especifica el tamaño máximo para rastreo de la XAPI, expresado en megabytes (definido aquí como "1048576 bytes"). Introduzca un número no negativo. El valor por defecto de esta opción es 50.

Se debe reiniciar el servidor de la XAPI para que se aplique esta variable.

XAPI_TRACE_FILE_NUM

Petición de datos: *Number of XAPI Trace archive files to retain [10]* (Número de archivos de almacenamiento de rastreo de la XAPI que se conservan [10]):

Esta opción especifica el número de archivos de rastreo de la XAPI almacenados que se deben conservar. Cuando el tamaño de `vtrace.file` excede el tamaño máximo, se cambia el sufijo del nombre del archivo de rastreo de 0 a n . El archivo más reciente es el que tiene el 0; el que tiene n es el más antiguo. Los archivos almacenados se guardan en el directorio `XAPI_WORK_PATH`. Cuando se alcanza el número especificado de logs almacenados, el archivo más antiguo se elimina del directorio de almacenamiento cada vez que se agrega uno nuevo a ese directorio. Puede conservar entre uno y 99 archivos almacenados. Introduzca un número entre 1 y 99 para especificar el número de archivos log almacenados que desea conservar.

Se debe reiniciar el servidor de la XAPI para que se aplique esta variable.

XAPI_STARTUP_FILE

Petición de datos: *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]* (Los cambios que se hagan en el nombre de archivo de inicio de la XAPI no tienen efecto hasta que se reinicie el servidor de la XAPI. Nombre del archivo de inicio de la XAPI con parámetros de control [xapi_startup_file]):

Esta opción especifica el nombre del archivo de inicio de la XAPI. El archivo reside en el directorio `XAPI_WORK_PATH` e incluye los parámetros de inicio de la XAPI.

Se debe reiniciar el servidor de la XAPI para que se aplique esta variable.

XAPI_TAPEPLEX_NAME

Petición de datos: *Changes to the XAPI Tapeplex name will not take effect until the xapi server is restarted. Name of the XAPI Tapeplex []* (Los cambios en el nombre de Tapeplex de

la XAPI no tienen efecto hasta que se reinicie el servidor. Nombre de Tapeplex de la XAPI []):

Esta opción especifica el nombre de Tapeplex de la XAPI. Introduzca un texto que no tenga más de 8 caracteres. Se debe reiniciar el servidor de la XAPI para que se aplique esta variable.

Índice

C

- comando DEFINE POOL_Access, 22
- comando DEFINE POOL_Name, 21
- comando DELETE POOL_Access, 25
- comando DELETE POOL_Name, 24
- comando LIST, 35
- comando LOG, 33
- comando MSGLVL, 37
- comando QUERY POOL_Access, 30
- comando QUERY POOL_Name, 29
- comando SET POOL_Name, 26
- comando TRACE, 38
- comandos
 - comandos de configuración y administración de XAPI, 21
 - comandos del operador del servidor de la XAPI, 33
 - 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, 38
- comandos de administración, 21
- comandos de configuración, 21
- comandos del operador, 33
- comandos, convenciones, 19
- convenciones de comandos, 19

D

- descripción
 - componente de interfaz XCMD, 17
 - interfaz del cliente XAPI para el servidor ACSLS, 17

E

- ejemplos

- comando DELETE POOL_Name, 25
- comando DELETE POOL_Access, 26
- comando LOG, 35, 37
- comando MSGLVL, 38
- comando QUERY POOL_Name, 30
- comando QUERY POOL_Access, 31
- comando Route, 44
- comando TRACE, 40
- DEFINE POOL_Access, 22, 24
- SET POOL_Name, 29

I

- introducción, 17
- introducción de comandos
 - desde cmd-proc de ACSLS, 19
 - uso del comando route del cliente de VM o SMC, 44

S

- sintaxis
 - comando DEFINE POOL_Name, 21
 - comando DEFINE POOL_Access, 23
 - comando DELETE POOL_Name, 24
 - comando DELETE POOL_Access, 26
 - comando LOG, 33, 35
 - comando MSGLVL, 37
 - comando QUERY POOL_Name, 29
 - comando QUERY POOL_Access, 30
 - comando SET POOL_Name, 27
 - comando TRACE, 39

V

- variables de control de la XAPI de ACSLS, 45
- variables, ACSLS, 45

