

StorageTek Enterprise Library Software

ACSL S 서버에 대한 XAPI 클라이언트 인터페이스 참조

E74277-01

2015년 9월

StorageTek Enterprise Library Software

ACSL S 서버에 대한 XAPI 클라이언트 인터페이스 참조

E74277-01

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

본 소프트웨어와 관련 문서는 사용 제한 및 기밀 유지 규정을 포함하는 라이선스 합의서에 의거해 제공되며, 지적 재산법에 의해 보호됩니다. 라이선스 합의서 상에 명시적으로 허용되어 있는 경우나 법규에 의해 허용된 경우를 제외하고, 어떠한 부분도 복사, 재생, 번역, 방송, 수정, 라이선스, 전송, 배포, 진열, 실행, 발행 또는 전시될 수 없습니다. 본 소프트웨어를 리버스 엔지니어링, 디스어셈블리 또는 디컴파일하는 것은 상호 운용에 대한 법규에 의해 명시된 경우를 제외하고는 금지되어 있습니다.

이 안의 내용은 사전 공지 없이 변경될 수 있으며 오류가 존재하지 않음을 보증하지 않습니다. 만일 오류를 발견하면 서면으로 통지해 주시기 바랍니다.

만일 본 소프트웨어나 관련 문서를 미국 정부나 또는 미국 정부를 대신하여 라이선스한 개인이나 법인에게 배송하는 경우, 다음 공지 사항이 적용됩니다.

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.

본 소프트웨어 혹은 하드웨어는 다양한 정보 관리 애플리케이션의 일반적인 사용을 목적으로 개발되었습니다. 본 소프트웨어 혹은 하드웨어는 개인적인 상해를 초래할 수 있는 애플리케이션을 포함한 본질적으로 위험한 애플리케이션에서 사용할 목적으로 개발되거나 그 용도로 사용될 수 없습니다. 만일 본 소프트웨어 혹은 하드웨어를 위험한 애플리케이션에서 사용할 경우, 라이선스 사용자는 해당 애플리케이션의 안전한 사용을 위해 모든 적절한 비상-안전, 백업, 대비 및 기타 조치를 반드시 취해야 합니다. Oracle Corporation과 그 자회사는 본 소프트웨어 혹은 하드웨어를 위험한 애플리케이션에서의 사용으로 인해 발생하는 어떠한 손해에 대해서도 책임지지 않습니다.

Oracle과 Java는 Oracle Corporation 및/또는 그 자회사의 등록 상표입니다. 기타의 명칭들은 각 해당 명칭을 소유한 회사의 상표일 수 있습니다.

Intel 및 Intel Xeon은 Intel Corporation의 상표 내지는 등록 상표입니다. SPARC 상표 일체는 라이선스에 의거하여 사용되며 SPARC International, Inc.의 상표 내지는 등록 상표입니다. AMD, Opteron, AMD 로고 및 AMD Opteron 로고는 Advanced Micro Devices의 상표 내지는 등록 상표입니다. UNIX는 The Open Group의 등록상표입니다.

본 소프트웨어 혹은 하드웨어와 관련문서(설명서)는 제3자로부터 제공되는 콘텐츠, 제품 및 서비스에 접속할 수 있거나 정보를 제공합니다. 사용자와 오라클 간의 합의서에 별도로 규정되어 있지 않는 한 Oracle Corporation과 그 자회사는 제3자의 콘텐츠, 제품 및 서비스와 관련하여 어떠한 책임도 지지 않으며 명시적으로 모든 보증에 대해서도 책임을 지지 않습니다. Oracle Corporation과 그 자회사는 제3자의 콘텐츠, 제품 및 서비스에 접속하거나 사용으로 인해 초래되는 어떠한 손실, 비용 또는 손해에 대해 어떠한 책임도 지지 않습니다. 단, 사용자와 오라클 간의 합의서에 규정되어 있는 경우는 예외입니다.

차례

머리말	9
대상	9
설명서 접근성	9
관련 문서	9
규약	9
구문 표기법	10
플로우 라인	10
단일 필수 선택	10
단일 옵션 선택	10
기본값	11
반복	11
키워드	11
변수	11
대체	11
선택사항	11
구분자	12
범위	12
16진수 범위	12
10진수 범위	12
HSC 형식 Volser 범위	12
ACSLs 형식 영숫자 Volser 범위	13
목록	14
공백	14
1. 소개	15
ACSLs 서버에 대한 XAPI 클라이언트 인터페이스란?	15
XCMD 인터페이스 구성 요소란?	15
2. XCMD 명령 규약	17
3. XCMD XAPI 구성 및 관리 명령	19
DEFINE POOL_Name	19
설명	19

구문	19
매개변수	20
예	20
DEFINE POOL_Access	20
설명	20
구문	20
매개변수	21
예	22
DELETE POOL_Name	22
설명	22
구문	22
매개변수	22
예	23
DELETE POOL_Access	23
설명	23
구문	23
매개변수	23
예	24
SET POOL_Name	24
설명	24
구문	24
매개변수	25
예	26
QUERY POOL_Name	26
설명	26
구문	26
매개변수	27
예	27
QUERY POOL_Access	27
설명	27
구문	28
매개변수	28
예	28
4. XCMD XAPI 서버 운영자 명령	29
LOG	29
설명	29
구문	29
매개변수	30

예	30
LIST	31
설명	31
구문	31
매개변수	31
예	33
MSGLVL	33
설명	33
구문	33
매개변수	33
예	34
TRACE	34
설명	34
구문	34
매개변수	35
예	35
5. XAPI 사용자 요청 및 HSC/VTCS 명령에 대한 XCMD 지원	37
지원되는 XAPI 요청, 명령 및 매개변수	37
SMC 또는 VM 클라이언트 경로 지정 명령을 사용해서 ACSLS XAPI 서버에 대 한 명령 실행	40
6. ACSLS XAPI 서버 제어 변수	41
ACSLX XAPI 서버 제어 변수 설정	41
ACSLX XAPI 변수 설명	42
XAPI_PORT	42
XAPI_WORK_PATH	42
XAPI_LOG_SIZE	43
XAPI_LOG_FILE_NUM	43
XAPI_TRACE_SIZE	43
XAPI_TRACE_FILE_NUM	43
XAPI_STARTUP_FILE	44
XAPI_TAPEPLEX_NAME	44
색인	45

표 목 록

5.1. XCMD 지원 XAPI 요청 및 해당 HSC/MVS 명령	37
--	----

머리말

이 문서에서는 XAPI 지원이 설치된 Oracle StorageTek ACSLS 서버에 대한 구성 및 관리 정보를 제공합니다.

대상

이 문서는 스토리지 관리자, 시스템 프로그래머 및 ACSLS XAPI 서버 구성 및 유지 관리 책임이 있는 운영자를 대상으로 합니다.

설명서 접근성

오라클의 접근성 개선 노력에 대한 자세한 내용은 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>에서 Oracle Accessibility Program 웹 사이트를 방문하십시오.

오라클 고객지원센터 액세스

지원 서비스를 구매한 오라클 고객은 My Oracle Support를 통해 온라인 지원에 액세스할 수 있습니다. 자세한 내용은 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>를 참조하거나, 청각 장애가 있는 경우 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>를 방문하십시오.

관련 문서

자세한 내용은 Oracle StorageTek ELS 릴리스 7.3 설명서 세트 및 ACSLS 릴리스 8.4 설명서 세트에서 다음 문서를 참조하십시오.

- *ACSLS 8.4 Administrator's Guide*
- *ACSLS 8.4 Installation Guide*
- ACSLS 8.4 빠른 참조
- *ELS 7.3 Command, Control Statement, and Utility Reference*
- *Configuring and Managing SMC 7.3*
- *VM Client* 설치, 구성 및 관리 설명서

규약

이 문서에 사용된 텍스트 규약은 다음과 같습니다.

규약	의미
굵은체	굵은체 유형은 작업과 연관된 그래픽 사용자 인터페이스 요소, 또는 텍스트나 용어집에 정의된 용어를 나타냅니다.

규약	의미
기울임꼴	기울임꼴 유형은 책 제목, 강조 또는 사용자가 특정 값을 제공할 위치 표시자 변수를 나타냅니다.
고정 폭	고정 폭 유형은 단락 안의 명령, URL, 예제의 코드, 화면에 나타나는 텍스트, 사용자가 입력한 텍스트를 나타냅니다.

구문 표기법

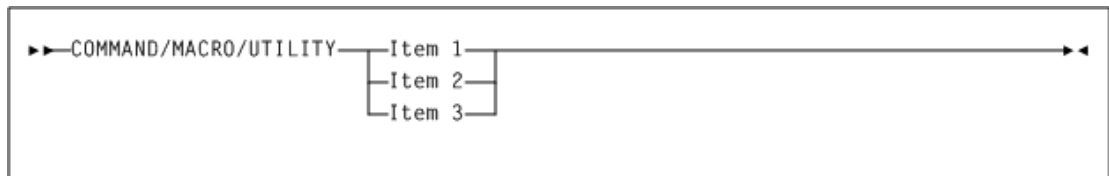
주:

다음 구문 규약은 XAPI 및 XCMD 구문에만 적용됩니다. ACSLS 명령 및 유틸리티에 적용되는 구문 규약에 대한 자세한 내용은 *ACSLs Administrator's Guide*를 참조하십시오.

구문 플로우 다이어그램 규약은 다음과 같습니다.

플로우 라인

구문 다이어그램은 수평 기준선, 수평 및 세로 분기 선, 명령, 제어문, 매크로 또는 유틸리티에 대한 텍스트로 구성됩니다. 다이어그램은 왼쪽에서 오른쪽, 위에서 아래의 순서로 읽습니다. 화살표는 플로우 및 방향을 나타냅니다. 예를 들면 다음과 같습니다.



단일 필수 선택

분기 선은 단일 옵션을 선택해야 함을 나타냅니다. 선택할 항목 중 하나가 다이어그램의 기준선에 배치되어 있으면 한 가지 항목을 선택해야 합니다. 예를 들면 다음과 같습니다.



단일 옵션 선택

첫번째 항목이 기준선 아래의 선에 배치되어 있으면, 하나의 항목을 옵션으로 선택할 수 있습니다. 예를 들면 다음과 같습니다.



기본값

기본값 및 매개변수는 기준선 아래에 표시됩니다. 예를 들면 다음과 같습니다.



일부 키워드 매개변수는 한 가지 스택의 값 선택 옵션을 제공합니다. 스택에 기본값이 포함된 경우, 키워드 및 값 옵션이 기준선 아래에 배치되어 선택사항임을 나타내고, 기본값은 키워드 선 위에 표시됩니다. 예를 들면 다음과 같습니다.



반복

반복 기호는 하나 이상의 옵션을 선택하거나 단일 옵션을 여러 번 선택할 수 있음을 나타냅니다. 다음 예제에서는 반복 구분자로逗가 필요함을 나타냅니다. 예를 들면 다음과 같습니다.



키워드

모든 명령 키워드는 전체 대문자 또는 혼합된 대소문자로 표시됩니다. 명령이 대소문자를 구분하지 않을 경우, 혼합된 대소문자는 축약형을 만들 때 소문자 문자를 생략할 수 있음을 의미합니다.

변수

기울임꼴 서체는 변수를 나타냅니다.

i

대체

바()_i는 대체 매개변수 값을 구분합니다.

선택사항

대괄호 []는 명령 매개변수가 선택사항임을 나타냅니다.

구분자

콤마(,), 세미콜론(;) 또는 기타 구분자가 구문 다이어그램의 한 가지 요소로 표시된 경우, 해당 문의 일부로 이를 입력해야 합니다.

범위

포함 범위는 동일한 길이 및 데이터 유형의 요소 쌍(대시로 연결됨)으로 표시됩니다. 첫 번째 요소는 반드시 두 번째 요소보다 작아야 합니다.

16진수 범위

16진수 범위는 16진수 숫자의 쌍으로 구성됩니다(예: 0A2-0AD 또는 000-0FC).

10진수 범위

10진수 범위는 10진수 숫자의 쌍으로 구성됩니다(예: 1-9 또는 010-094). 선행 0은 필수가 아닙니다. 10진수 부분은 증분적 범위입니다. 두 범위 요소의 증분적 부분에서는 문자 위치가 일치해야 하며, 첫 번째 요소의 비증분 문자는 두 번째 요소의 비증분 문자와 동일해야 합니다.

HSC 형식 Volser 범위

HSC 형식 숫자 VOLSER 범위(vol-range)는 1~6자리의 10진수 숫자를 포함하는 VOLSER 요소 쌍으로 구성됩니다(예: ABC012-ABC025 또는 X123CB-X277CB). 10진수 부분은 증분적 범위입니다. 다음과 같은 추가 제한 사항이 적용됩니다.

- 두 범위 요소의 증분적 부분에 대한 문자 위치는 서로 일치해야 합니다.
- 첫 번째 요소의 비증분 문자는 두 번째 요소의 비증분 문자와 동일해야 합니다.
- 범위 요소의 두 부분은 증분할 수 없습니다. 111AAA가 첫 번째 요소면, 두 번째 요소에 대해 112AAB를 지정할 수 없습니다.
- VOLSER 범위에 2개 이상의 10진수 부분이 포함된 경우, 어떠한 부분이라도 증분 범위로 유효합니다. 예를 들면 다음과 같습니다.
 - A00B00 - 지정할 수 있는 가장 큰 범위는 A00B00~A99B99입니다.
 - A0B0CC - 지정할 수 있는 가장 큰 범위는 A0B0CC~A9B9CC입니다.
 - 000XXX - 지정할 수 있는 가장 큰 범위는 000XXX~999XXX입니다.

HSC 형식의 알파벳 VOLSER 범위(vol-range)는 1~6자의 증분 부분을 포함하는 VOLSER 요소 쌍으로 구성됩니다(예: 000AAA-000ZZZ 또는 9AAA55-9ZZZ55). 이 부분은 증분적 범위입니다. 다음과 같은 추가 제한 사항이 적용됩니다.

- 두 범위 요소의 증분적 부분에 대한 문자 위치는 서로 일치해야 합니다.
- 첫 번째 요소의 비증분 문자는 두 번째 요소의 비증분 문자와 동일해야 합니다.
- 범위 요소의 두 부분은 증분할 수 없습니다. 111AAA가 첫 번째 요소인 경우 두 번째 요소에 112AAB를 지정할 수 없습니다.

- VOLSER 범위의 영문자 부분은 A~Z 문자로 정의됩니다. 다중 문자 시퀀스를 증분할 경우 각 문자는 Z로 증분됩니다. 예를 들어, ACZ는 AAA-AMM 범위에 속합니다. 예제는 다음과 같습니다.
 - A00A0-A99A0
VOLSERs A00A0~A09A0로 증분된 후 A10A0~A99A0로 증분됩니다.
 - 9AA9A-9ZZ9A
VOLSERs 9AA9A~9AZ9A로 증분된 후 9BA9A~9ZZ9A로 증분됩니다.
 - 111AAA-111ZZZ
VOLSERs 111AAA~111AAZ로 증분된 후 111ABA~111ZZZ로 증분됩니다.
 - 999AM8-999CM8
VOLSERs 999AM8~999AZ8로 증분된 후 999BA8~999CM8로 증분됩니다.
 - A3BZZ9-A3CDE9
VOLSERs A3BZZ9~A3CAA9로 증분된 후 A3CAB9~A3CDE9로 증분됩니다.
 - AAAAAA-AAACCC
VOLSERs AAAAAA~AAAAAZ로 증분된 후 AAAABA~AAACCC로 증분됩니다.
 - CCCN NN-DDDN NN
VOLSERs CCCN NN~CCCN NZ로 증분된 후 CCCNOA~DDDN NN으로 증분됩니다. 이 범위는 매우 큰 범위입니다.

영문자 VOLSER 범위에서 볼륨 수는 VOLSER 범위의 증분 부분에 있는 요소 수에 따라 달라집니다. 각 문자 위치에 있는 A~Z 범위에서 볼륨 수는 26에 증분되는 부분 수를 제곱하여 계산할 수 있습니다.

- A-Z는 26^1 또는 26개 볼륨과 동일합니다.
- AA-ZZ는 26^2 또는 676개 볼륨과 동일합니다.
- AAA-ZZZ는 26^3 또는 17,576개 볼륨과 동일합니다.
- AAAA-ZZZZ는 26^4 또는 456,976개 볼륨과 동일합니다.
- AAAAA-ZZZZZ는 26^5 또는 11,881,376개 볼륨과 동일합니다.
- AAAAAA-ZZZZZZ는 26^6 또는 308,915,776개 볼륨과 동일합니다.

ACSL S 형식 영숫자 Volser 범위

ACSL S 형식 영숫자 volser 범위는 위에 지정된 HSC 형식 볼륨 패턴 범위를 따를 필요가 없습니다. 다음과 같은 추가 제한 사항이 적용됩니다.

- 범위에 지정된 두 volser는 동일한 문자 수를 포함해야 합니다.
- 범위에서 첫번째 volser 요소는 두번째 volser 요소보다 반드시 작아야 합니다.
- 지정된 volser의 길이가 6자 미만이면 공백이 채워져서 6자로 맞춰진 것처럼 평가됩니다.

- ASCII 조합 시퀀스를 사용해서 평가할 때 첫번째 volser 요소보다 반드시 크거나 같고 두 번째 volser 요소보다 반드시 작거나 같은 volser만 범위에 포함됩니다. 예를 들어, volser 범위 123456-234567에는 volser 12345A가 포함됩니다.

주:

- 유효한 HSC 형식 volser 범위는 ACSLS 형식 volser 규칙에서 항상 유효합니다. 하지만 HSC 형식 volser 규칙을 사용해서 평가할 때는 유효한 ACSLS 형식 volser 범위가 유효하지 않을 수 있습니다.
 - volser 범위를 허용하는 대부분의 XCMD 명령에서 범위는 HSC volser 규칙을 따라야 합니다. 어느 형식이든 volser 범위를 허용하는 유일한 명령은 XCMD SET POOL_Name 명령입니다.
-

목록

목록은 하나 이상의 요소로 구성됩니다. 두 개 이상의 요소가 지정된 경우, 콤마 또는 공백을 사용해서 요소를 구분해야 하며, 전체 목록을 괄호로 묶어야 합니다.

공백

키워드 매개변수 및 값은 임의 개수의 공백으로 구분할 수 있습니다.

1장. 소개

이 장에서는 ACSLS에 대한 XAPI 클라이언트 인터페이스를 소개하고 XAPI를 포함하는 ACSLS 서버를 대상으로 지정하는 명령을 입력할 수 있게 해주는 XCMD 인터페이스에 대해 설명합니다.

ACSL S 서버에 대한 XAPI 클라이언트 인터페이스란?

XAPI(XML API)는 StorageTek 클라이언트와 서버가 TCP/IP 상에서 공통 프로토콜을 사용하여 통신할 수 있는 Oracle StorageTek API입니다.

이 XAPI가 도입되어, 이전에 실제 테이프 처리를 위해 MVS 기반 서버(Oracle StorageTek 호스트 소프트웨어 구성 요소)를 사용해야 했던 고객은 이제 ACSLS(XAPI 지원 사용)를 다음과 같이 사용할 수 있습니다.

- 이제 MVS의 SMC 클라이언트는 MVS/CSC 필요 없이 ACSLS 서버(XAPI 지원 가능)에서 실제 테이프를 요청할 수 있습니다.

자세한 내용은 ELS 설명서 *Configuring and Managing SMC*를 참조하십시오.

- 이제 VM 클라이언트는 ACSLS 서버(XAPI 지원 가능)에서 실제 테이프 서비스를 요청할 수 있습니다.

자세한 내용은 ELS 설명서 *VM Client Installation, Configuration, and Administration Guide*를 참조하십시오.

XAPI 서비스는 ACSLS 8.4 및 이후 릴리스에서 지원됩니다. ACSLS XAPI 서비스 설치 및 구성에 대한 자세한 내용은 *StorageTek ACSLS 8.4 Installation Guide*를 참조하십시오.

XCMD 인터페이스 구성 요소란?

ACSL S cmd_proc 구성 요소는 고객이 ACSLS 시스템 및 ACSLS에서 제어되는 라이브러리를 관리할 수 있게 해주는 ACSLS를 위한 명령줄 인터페이스입니다.

cmd_proc 사용에 대한 자세한 내용은 *StorageTek Automated Cartridge System Library Software Administrator's Guide*의 Overview 장을 참조하십시오. XCMD 명령은 ACSLS cmd_proc의 확장 버전이며, ACSLS XAPI 서버 구성 요소가 설치되고 활성 상태일 때 ACSLS XAPI 서버 구성 요소 사용자가 기존 ACSLS cmd_proc 구성 요소에서 제한된 개수의 XAPI 서버 사용자 및 관리자 요청을 입력할 수 있게 해줍니다.

XCMD 명령을 사용해서 ACSLS cmd_proc에서 입력할 수 있는 XAPI 서버 명령에 대한 자세한 내용은 [3장. XCMD XAPI 구성 및 관리 명령](#) 을 참조하십시오.

XCMD 명령을 사용해서 ACSLS cmd_proc에서 입력할 수 있는 XAPI 운영자 명령에 대한 자세한 내용은 [4장. XCMD XAPI 서버 운영자 명령](#) 을 참조하십시오.

또한 SMC/MVS *Route* 명령 또는 VM/Client *Route* 명령을 사용해서 MVS 또는 VM 호스트에서 XAPI 서버 운영자 명령, XCMD 구성 및 관리 명령, HSC/VTCS XAPI 명령을 실행할 수 있습니다. 하지만 특정 매개변수 제한 사항이 적용됩니다.

지원되는 HSC/VTCS 명령 및 기타 XAPI 요청과 매개변수 제한 사항에 대한 자세한 내용은 [5장. XAPI 사용자 요청 및 HSC/VTCS 명령에 대한 XCMD 지원](#)을 참조하십시오.

2장. XCMD 명령 규약

SMC/MVS와 VM/클라이언트 사이의 대부분의 클라이언트/서버 상호 작용 및 XAPI를 포함하는 ACSLS 서버는 최종 사용자에게 투명하게 처리됩니다. 볼륨 정보, 마운트 및 마운트 해제에 대한 요청은 SMC/MVS 및 VM/클라이언트에서 자동으로 생성되며, 운영자 개입 없이 처리됩니다.

이러한 자동 상호 작용 외에도 XAPI를 포함하는 ACSLS 서버는 사용자가 XAPI 구성 요소를 관리할 수 있게 해주는 추가 관리자, 구성 및 운영자 명령을 제공합니다.

- XCMD XAPI 서버 구성 및 관리 명령에 대한 자세한 내용은 [3장. XCMD XAPI 구성 및 관리 명령](#) 을 참조하십시오.
- XCMD XAPI 서버 운영자 명령에 대한 자세한 내용은 [4장. XCMD XAPI 서버 운영자 명령](#) 을 참조하십시오.

이러한 명령은 ACSLS 기본 설치에 포함되지 않지만, ACSLS XAPI 구성 요소가 선택적으로 설치된 경우 사용할 수 있습니다. ACSLS 및 ACSLS XAPI 구성 요소 설치에 대한 자세한 내용은 *StorageTek Automated Cartridge System Library (ACSL) Installation Guide*를 참조하십시오.

이러한 명령을 실행하려면 ACSLS cmd_proc에서 실행된 XCMD 명령 기능을 사용합니다. 이 명령의 일반 구문은 다음과 같습니다.

XCMD command

여기서 *command*는 이 설명서에 설명된 ACSLS XAPI 구성 요소 명령 중 하나입니다.

주:

또한 XCMD 명령 기능을 사용해서 ACSLS XAPI 서버에서 지원되는 XAPI 사용자 요청을 실행할 수 있습니다. 자세한 내용은 [5장. XAPI 사용자 요청 및 HSC/VTCS 명령에 대한 XCMD 지원](#)을 참조하십시오.

다음 규약은 ACSLS cmd-proc에서 XCMD 명령을 실행할 때마다 적용됩니다.

- XCMD는 ACSLS 명령 프롬프트를 사용하지 않습니다.
- 연관된 값과 함께 키워드 매개변수를 입력할 때, 값은 단일 공백 문자로 시작해야 합니다.
- XCMD 명령은 대문자 또는 소문자로 입력할 수 있습니다. ACSLS XAPI 구성 요소는 모든 명령 입력을 자동으로 대문자로 변환합니다. 예를 들어, 하위 풀 이름

"Testpool1", "testpool1", "TestPool1" 및 "TESTPOOL1"은 모두 동일한 하위 풀 이름인 "TESTPOOL1"로 변환됩니다.

주:

일반 ACSLS cmd_proc 명령 및 매개변수는 일반적으로 소문자로 지정됩니다. vol_ids(volsers)를 제외하고는 소문자가 대문자로 자동으로 변환되지 않습니다.

-
- XCMD 매개변수 값은 같음 기호(=)로 시작할 수 없습니다.
 - XCMD 값은 괄호로 묶을 수 없습니다. 여러 값이 지원될 경우에는逗를 사용하여 각 값을 구분할 수 있습니다. 예를 들어, 여러 값을 질의하려면 XCMD QUERY VOLUME(volser1,volser2)이 아니라 XCMD QUERY VOLUME volser1,volser2를 입력합니다.

3장. XCMD XAPI 구성 및 관리 명령

이 장에서는 XCMD에서 지원되는 XAPI 구성 및 관리 명령에 대해 설명합니다. 이러한 명령은 ACSLS 릴리스 8.4에서 ELS 호환 볼륨 풀 기능을 사용으로 설정합니다. 지원되는 명령은 다음과 같습니다.

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

XCMD 명령을 사용해서 ACSLS user_proc에서 이러한 명령을 실행합니다. 자세한 내용은 [2장. XCMD 명령 규약](#) 을 참조하십시오.

다음 명령 섹션에 포함된 구문 플로우 다이어그램에 사용되는 규약은 "[구문 표기법](#)"을 참조하십시오.

DEFINE POOL_Name

다음 절에서는 *DEFINE POOL_Name* 명령에 대해 설명합니다.

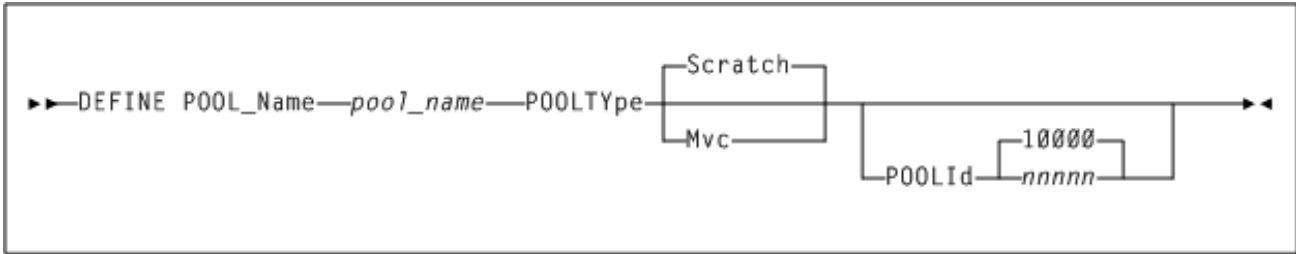
설명

DEFINE POOL_NAME 명령은 풀 유형 및 풀 이름을 ACSLS 풀 ID와 연결합니다.

구문

다음 그림은 *DEFINE POOL_Name* 명령의 구문을 보여줍니다.

그림 3.1. DEFINE POOL_Name 명령 구문



매개변수

그림 3.1. “DEFINE POOL_Name 명령 구문”에 표시된 것처럼 `DEFINE POOL_Name` 명령에는 다음 매개변수가 포함됩니다.

pool_name

최대 13자의 풀 이름. 풀 이름은 퍼센트 기호("%"), 밑줄("_") 또는 별표("*")를 제외한 모든 문자를 포함할 수 있습니다.

POOLType

`Scratch`(기본값) 또는 `Mvc`의 풀 유형.

POOLId(nnnnn)

선택적으로, 풀 ID(`nnnnn`)를 지정합니다.

- 이 매개변수를 지정하도록 선택한 경우, 현재 존재하지 않는 풀 ID를 지정해야 합니다.
- 이 매개변수를 지정하지 않으면 시스템이 10000부터 시작해서 사용되지 않은 첫번째 값으로 풀 ID를 자동으로 지정합니다.

예

다음 예에서 사용자는 `DEFINE POOL_Name` 명령을 실행해서 스크래치 풀 `SCRPOOL1`을 정의합니다.

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

DEFINE POOL_Access

다음 절에서는 `DEFINE POOL_Access` 명령에 대해 설명합니다.

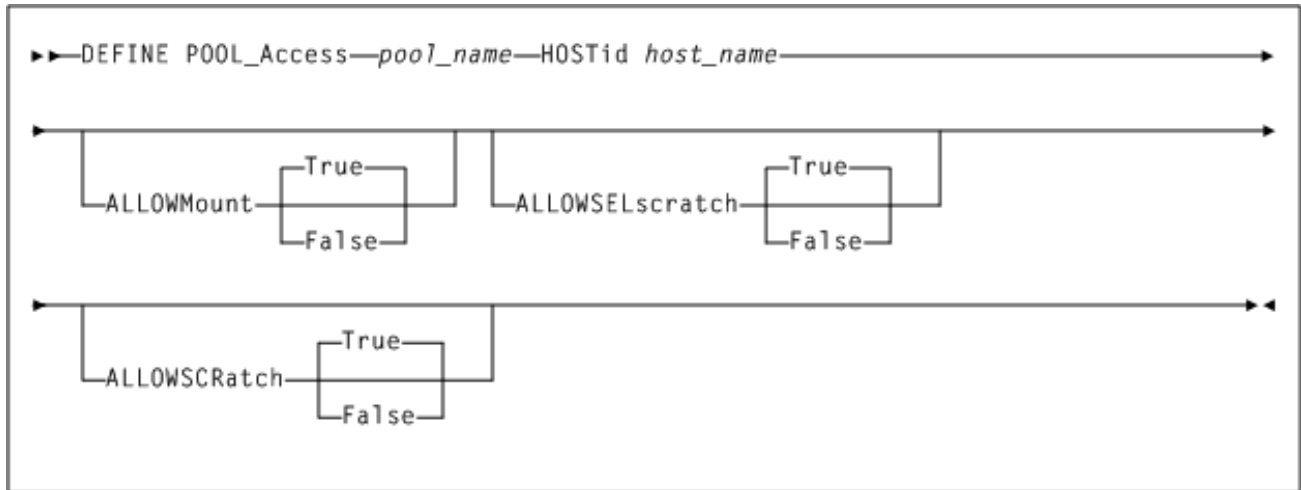
설명

`DEFINE POOL_Access` 명령은 지정된 호스트에서 `pool_name`에 대해 허용되는 액세스를 정의합니다. 특정 `pool_name`에 대해 풀 액세스가 정의되지 않은 경우 모든 액세스가 허용됩니다.

구문

다음 그림은 `DEFINE POOL_Access` 명령의 구문을 보여줍니다.

그림 3.2. DEFINE POOL_Access 명령 구문



매개변수

그림 3.2. “DEFINE POOL_Access 명령 구문”에 표시된 것처럼 `DEFINE POOL_Access` 명령에는 다음 매개변수가 포함됩니다.

pool_name

최대 13자의 풀 이름. 풀 이름은 퍼센트 기호("%"), 밑줄("_") 또는 별표("*")를 제외한 모든 문자를 포함할 수 있습니다. 지정된 `pool_name`은 `SCRATCH` 풀 유형과 함께 `DEFINE POOL_Name` 명령을 사용해서 이전에 정의되어 있어야 합니다(MVC 풀에 대해 액세스가 정의되지 않음). 자세한 내용은 "[DEFINE POOL_Name](#)"을 참조하십시오.

HOSTid host_name

최대 31자의 호스트 이름을 지정합니다. 이 매개변수는 필수입니다.

지정된 `HOSTid`는 인터넷 전체 도메인 이름 중에서 호스트 이름 부분입니다. 예를 들어, 전체 도메인 이름이 `myhost.plus.domain.name`인 경우, 지정된 `HOSTid`는 `myhost`입니다.

ALLOWMount

선택적으로, 마운트 액세스를 지정합니다. `True` 또는 `False`를 지정할 수 있습니다. 이 매개변수는 `POOLTYPE MVC`로 정의된 풀에 대해서만 유효합니다.

ALLOWSELscratch

선택적으로, 스크래치 볼륨에 대해 XAPI 명령 `query_scr_mnt_info`, `sel_scratch` 및 `mount`에 대한 액세스를 지정합니다. `True` 또는 `False`를 지정할 수 있습니다. 이 매개변수를 지정하지 않으면 기본적으로 `True`로 설정됩니다.

이 매개변수는 `POOLTYPE SCRATCH`로 정의된 풀에 대해서만 유효합니다. `MVC`로 정의된 풀에 대해 이 매개변수를 지정할 경우 매개변수가 자동으로 기본값 `False`로 설정됩니다.

ALLOWSCRATCH

선택적으로, 스크래치 액세스를 지정합니다. *True* 또는 *False*를 지정할 수 있습니다. 이 매개변수를 지정하지 않으면 기본적으로 *True*로 설정됩니다.

이 매개변수는 *POOLTYPE SCRATCH*로 정의된 풀에 대해서만 유효합니다. *MVC*로 정의된 풀에 대해 이 매개변수를 지정할 경우 매개변수가 자동으로 기본값 *False*로 설정됩니다.

예

다음 예에서 사용자는 *DEFINE POOL_Access* 명령을 실행해서 스크래치 풀 이름 *SCRPOOL1*에 대해 *set access*를 정의합니다. 여기서 호스트는 *PRODMVS*이고 *ALLOWMOUNT* 및 *ALLOWSCRATCH*는 *True*입니다.

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

DELETE POOL_Name

다음 절에서는 *DELETE POOL_Name* 명령에 대해 설명합니다.

설명

DELETE POOL_Name 명령은 풀을 삭제합니다. 이 명령은 *ACSL delete pool* 명령과 비슷하지만 다음과 같은 예외사항이 있습니다.

- 입력은 *pool ID* 대신 *pool name*을 지정합니다.
- 사용자는 풀에 지정된 볼륨을 풀 0에 다시 지정할 수 있습니다.

ACSL delete pool 명령에 대한 자세한 내용은 *StorageTek Automated Cartridge System Library Software (ACSL) Administrator's Guide*를 참조하십시오.

구문

다음 그림은 *DELETE POOL_Name* 명령의 구문을 보여줍니다.

그림 3.3. DELETE POOL_Name 명령 구문

**매개변수**

그림 3.3. “*DELETE POOL_Name* 명령 구문”에 표시된 것처럼 *DELETE POOL_Name* 명령에는 다음 매개변수가 포함됩니다.

pool_name

최대 13자의 풀 이름. 풀 이름은 퍼센트 기호("%"), 밑줄("_") 또는 별표("*")를 제외한 모든 문자를 포함할 수 있습니다. 지정된 *pool_name*은 *DEFINE POOL_Name* 명령을 사용해서 이전에 정의되어 있어야 합니다. 자세한 내용은 "[DEFINE POOL_Name](#)"을 참조하십시오.

RESETvols

삭제하려는 풀에 현재 지정된 볼륨을 풀 0에 다시 지정해야 함을 나타냅니다. 이 매개변수를 지정하지 않고, 볼륨이 현재 풀 내에 존재하면 *delete pool-name* 명령이 실패합니다.

예

다음 예에서 사용자는 *DELETE POOL_Name* 명령을 실행해서 *SCRPOOL1* 풀을 삭제하고 해당 볼륨을 풀 0(기본 풀)에 다시 지정합니다.

```
XCMD DELETE POOL_NAME SCRPOOL_TEST1 RESETVOLS
```

DELETE POOL_Access

다음 절에서는 *DELETE POOL_Access* 명령에 대해 설명합니다.

설명

DELETE POOL_Access 명령은 단일 호스트 이름(*host_id*)과 연관된 풀 액세스 규칙을 삭제하거나 모든 호스트에 대한 모든 풀 액세스 규칙을 삭제합니다.

- 호스트 이름(*host_id*)이 지정된 경우 지정된 호스트에 대한 액세스 규칙만 지정된 풀에서 삭제됩니다.
- *ALLhost*가 지정된 경우 해당 풀의 모든 풀 액세스 규칙이 삭제됩니다.

구문

다음 그림은 *DELETE POOL_Access* 명령의 구문을 보여줍니다.

그림 3.4. DELETE POOL_Access 명령 구문

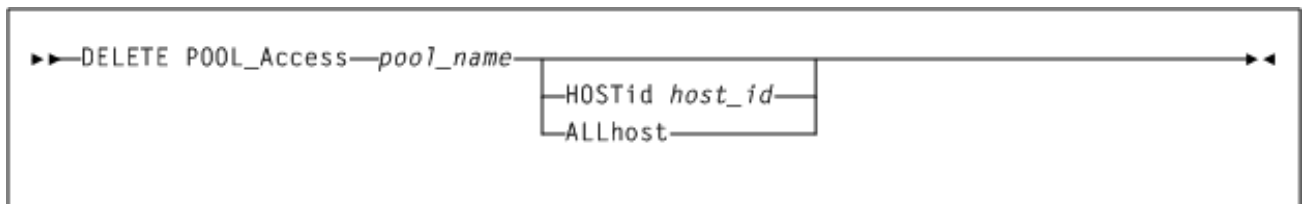
**매개변수**

그림 3.4. "[DELETE POOL_Access 명령 구문](#)"에 표시된 것처럼 *DELETE POOL_Access* 명령에는 다음 매개변수가 포함됩니다.

pool_name

최대 13자의 풀 이름. 풀 이름은 퍼센트 기호("%"), 밑줄("_") 또는 별표("*")를 제외한 모든 문자를 포함할 수 있습니다. 지정된 *pool_name*은 *DEFINE POOL_Name* 명령을 사용해서 이전에 정의되어 있어야 합니다. 자세한 내용은 "*DEFINE POOL_Name*"을 참조하십시오.

HOSTid host_name

선택적으로, 최대 31자의 호스트 이름을 지정하고 이 호스트 ID에 대한 액세스 제한을 제거해야 함을 나타냅니다(변경되지 않은 상태의 다른 호스트 ID 액세스 규칙 사용). 이 매개변수는 *ALLhost*와 상호 배타적입니다.

ALLhost

선택적으로, 모든 호스트 ID에 대한 특정 *pool_name*의 모든 액세스 제한이 제거되도록 지정합니다. 이 매개변수는 *HOSTid*와 상호 배타적입니다.

예

다음 예에서 사용자는 *DELETE POOL_Access* 명령을 실행해서 *PRODMVS* 호스트에서 *SCRPOOL1*로의 풀 액세스를 삭제합니다.

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

SET POOL_Name

다음 절에서는 *SET POOL_Name* 명령에 대해 설명합니다.

설명

SET POOL_Name 명령은 볼륨을 풀 이름에 지정하거나 풀 이름에 대해 하위 워터마크를 설정합니다.

주:

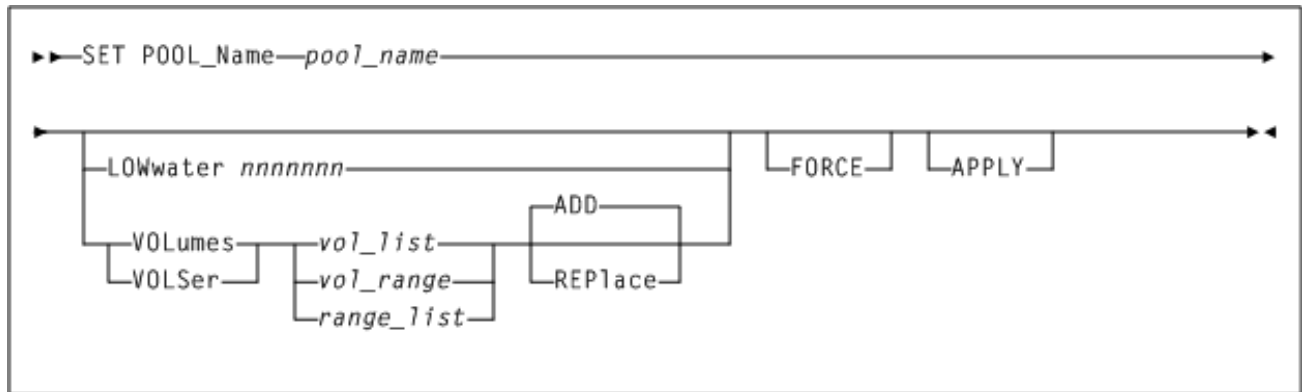
명령 구문이 유효한 경우, 범위 내 일부 볼륨을 풀에 추가할 수 없더라도 지정된 볼륨 범위가 데이터베이스에 추가됩니다. 다음 중 하나가 true인 경우 볼륨을 풀에 추가할 수 없습니다.

- 볼륨이 *LOCK*되어 있습니다.
- *FORCE*가 지정되지 않은 한 볼륨이 이미 다른 풀에 있습니다.
- 볼륨이 논리적 라이브러리에 있습니다.
- 볼륨에 *ACSL*S 소유자가 포함되며 액세스 제어가 사용됩니다.

구문

다음 그림은 *SET POOL_Name* 명령의 구문을 보여줍니다.

그림 3.5. SET POOL_Name 명령 구문



매개변수

그림 3.5. “SET POOL_Name 명령 구문”에 표시된 것처럼 SET POOL_Name 명령에는 다음 매개변수가 포함됩니다.

pool_name

최대 13자의 풀 이름. 풀 이름은 퍼센트 기호("%"), 밑줄("_") 또는 별표("*")를 제외한 모든 문자를 포함할 수 있습니다. 지정된 pool_name은 DEFINE POOL_Name 명령을 사용해서 이전에 정의되어 있어야 합니다. 자세한 내용은 “DELETE POOL_Name”을 참조하십시오.

LOWwater nnnnnnn

선택적으로, 풀이 포함해야 하는 사용 가능한 볼륨의 최소 개수를 정의합니다. 사용 가능한 볼륨 수가 하위 워터마크 아래로 떨어지면 ACSLS 시스템이 경고 메시지를 생성합니다.

VOLumes 또는 VOLSer(volser-list, volser-range, range-list)

선택적으로, volser 목록, volser 범위 또는 volser 범위 목록을 지정합니다. 최대 16개의 단일 볼륨 또는 범위를 지정할 수 있습니다.

볼륨 범위는 ACSII 조합 시퀀스인 ACSLS 형식 volser 범위에 따라 평가되며, HSC 볼륨 패턴 범위를 따를 필요가 없습니다. 하지만 단일 범위의 볼륨 일련 번호는 동일한 문자 수를 포함해야 하며, 일치하는 문자 수의 볼륨 일련 번호만 범위 내에 있는 것으로 간주됩니다. 예를 들어, 1234-12356 범위는 잘못된 범위입니다. 1234-2345 범위에서 볼륨 일련 번호 123567은 범위 내에 있는 것으로 고려되지 않습니다. 지정된 범위가 유효한 경우 범위 내 일부 볼륨을 풀에 지정할 수 없더라도 데이터베이스에 추가됩니다. 지정된 볼륨 범위는 기존 볼륨 범위와 겹치지 않아야 하며 서로 겹치지 않아야 합니다.

볼륨 범위는 ACSLS 형식 volser 범위에 따라 평가됩니다.

- 범위에 지정된 두 volser는 동일한 문자 수를 포함해야 합니다.
- 범위에서 첫번째 volser 요소는 두번째 volser 요소보다 반드시 작아야 합니다.
- 지정된 volser의 길이가 6자 미만이면 공백이 채워져서 6자로 맞춰진 것처럼 평가됩니다.

- ASCII 조합 시퀀스를 사용해서 평가할 때 첫번째 volser 요소보다 반드시 크거나 같고 두번째 volser 요소보다 반드시 작거나 같은 volser만 범위에 포함됩니다. 예를 들어, volser 범위 123456-234567에는 volser 12345A가 포함됩니다.

지정된 범위가 유효하면 범위 내 일부 볼륨을 풀에 지정할 수 없더라도 *pool_name* 데이터베이스에 추가됩니다. 지정된 볼륨 범위는 기존 볼륨 범위와 겹치지 않아야 하며 서로 겹치지 않아야 합니다.

ADD

풀의 기존 범위에 지정된 볼륨 범위를 추가해야 함을 나타냅니다. 이 옵션은 기본값입니다. 지정된 경우 *VOLumes*도 지정해야 합니다. *ADD*는 *REPLace*와 상호 배타적입니다.

REPLace

지정된 볼륨 범위를 현재 지정된 볼륨 범위로 바꾸고, 이전 범위의 볼륨을 풀 0에 다시 지정해야 함을 나타냅니다. 지정된 경우 *VOLumes*(또는 *VOLSer*)도 지정해야 합니다. *REPLace*는 *ADD*와 상호 배타적입니다.

FORCE

선택적으로, 다른 풀에 이미 지정된 볼륨을 새 풀에 다시 지정해야 함을 나타냅니다. 이 매개변수가 지정되지 않은 경우 현재 다른 풀에 지정된 볼륨은 업데이트되지 않습니다. 지정된 경우 *VOLumes*(또는 *VOLSer*)도 지정해야 합니다.

APPLY

선택적으로, 이전에 정의된 볼륨 범위를 데이터베이스의 볼륨에 적용해야 함을 나타냅니다. 이 매개변수는 위에 나열된 제한 사항으로 인해 일부 볼륨이 업데이트되지 않았을 때 사용할 수 있습니다. 지정된 경우 *VOLumes*(또는 *VOLSer*)도 지정해야 합니다.

예

다음 예에서 명령은 *DRL001-DRL999* 볼륨을 *SCRPOOL1*에 추가합니다.

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

QUERY POOL_Name

다음 절에서는 *QUERY POOL_Name* 명령에 대해 설명합니다.

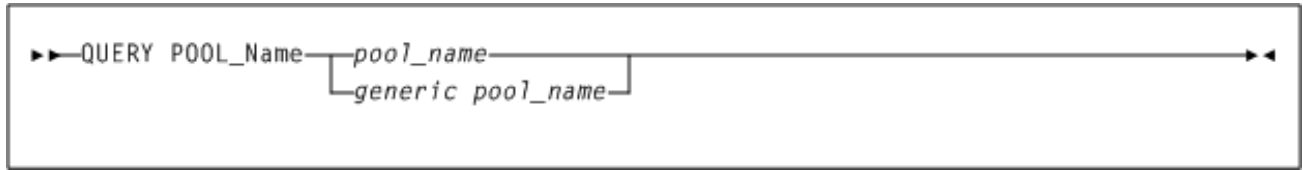
설명

QUERY POOL_Name 명령은 모든 풀, 지정된 문자로 시작하는 풀 또는 단일 풀을 표시합니다.

구문

다음 그림은 *QUERY POOL_Name* 명령의 구문을 보여줍니다.

그림 3.6. QUERY POOL_Name 명령 구문



매개변수

그림 3.6. “QUERY POOL_Name 명령 구문”에 표시된 것처럼 *QUERY POOL_Name* 명령에는 다음 매개변수가 포함됩니다.

pool_name

최대 13자의 기존 풀 이름 또는 마지막 별표(*) 문자가 있는 일반 이름입니다. 이 매개변수는 필수입니다.

pool_name 형식은 다음과 같이 출력 표시를 결정합니다.

- *pool_name*이 일반이 아니면, 표시된 풀 정보에 일치하는 단일 풀에 대해 정의된 volser 및 volser 범위가 포함됩니다.
- *pool_name*이 일반인 경우(후행 '*' 포함), 표시된 풀 정보에는 일치하는 풀에 대해 정의된 volser 및 volser 범위가 포함되지 않습니다.

예

다음 예에서 사용자는 *QUERY POOL_Name* 명령을 실행해서 연관된 볼륨 범위를 포함하여 pool1에 대한 정보를 표시합니다.

```
XCMD QUERY POOL_NAME POOL1
```

다음 예에서 사용자는 *QUERY POOL_Name*을 실행해서 이름 지정된 모든 풀에 대한 정보를 표시합니다(볼륨 범위 포함 안함).

```
XCMD QUERY POOL_NAME *
```

다음 예에서 사용자는 *QUERY POOL_Name* 명령을 실행해서 "A"로 시작하는 모든 풀에 대한 정보를 표시합니다(볼륨 범위 포함 안함).

```
XCMD QUERY POOL_NAME A*
```

QUERY POOL_Access

다음 절에서는 *QUERY POOL_Access* 명령에 대해 설명합니다.

설명

QUERY POOL_Access 명령은 단일 호스트 또는 풀 이름과 연관된 모든 호스트에 대한 풀 액세스 규칙을 표시합니다.

구문

다음 그림은 `QUERY POOL_Access` 명령의 구문을 보여줍니다.

그림 3.7. `QUERY POOL_Access` 명령 구문



매개변수

그림 3.7. “`QUERY POOL_Access` 명령 구문”에 표시된 것처럼 `QUERY POOL_Access` 명령에는 다음 매개변수가 포함됩니다.

pool_name

최대 13자의 풀 이름. 풀 이름은 퍼센트 기호("%"), 밑줄("_") 또는 별표("*")를 제외한 모든 문자를 포함할 수 있습니다. 지정된 `pool_name`은 `DEFINE POOL_Name` 명령을 사용해서 이전에 정의되어 있어야 합니다. 자세한 내용은 "[DEFINE POOL_Name](#)"을 참조하십시오.

HOSTid host_id

선택적으로, 1-31자의 `host_name`을 지정합니다. 지정된 `host_id`는 `DEFINE POOL_Access` 명령을 사용해서 이전에 `pool_name`에 대해 정의되어 있어야 합니다. 자세한 내용은 "[DEFINE POOL_Access](#)"를 참조하십시오.

이 매개변수를 지정하지 않으면 풀과 연관된 모든 호스트에 대한 액세스 규칙이 표시됩니다.

예

다음 예에서 사용자는 `QUERY POOL_Access` 명령을 실행해서 `pool_name SCRPOOL1`과 연관된 모든 호스트에 대한 액세스 규칙을 질의합니다.

```
XCMD QUERY POOL_ACCESS SCRPOOL1
```

4장. XCMD XAPI 서버 운영자 명령

이 장에서는 XCMD에서 지원되는 XAPI 서버 운영자 명령에 대해 설명합니다. 이러한 명령은 운영자 또는 관리자가 ACSLS XAPI 구성 요소를 모니터링할 수 있게 해줍니다. 지원되는 명령은 다음과 같습니다.

- LOG
- LIST
- MSGLVL
- TRACE

XCMD 명령을 사용해서 ACSLS user_proc에서 이러한 명령을 실행합니다. 자세한 내용은 [2장. XCMD 명령 규약](#) 을 참조하십시오.

다음 명령 섹션에 포함된 구문 플로우 다이어그램에 사용되는 규약은 "[구문 표기법](#)"을 참조하십시오.

LOG

다음 절에서는 LOG 명령에 대해 설명합니다.

설명

LOG 명령은 현재 XAPI 서버 로그 설정을 변경 또는 나열합니다.

주:

로그 파일 이름 및 위치는 ACSLS 환경 변수로 설정되며 기본적으로 XAPI_WORK_PATH 디렉토리에서 파일 이름 *vlog.file*로 지정됩니다.

구문

다음 그림은 LOG 명령의 구문을 보여줍니다.

그림 4.1. LOG 명령 구문



매개변수

그림 4.1. “LOG 명령 구문”에 표시된 것처럼 *LOG* 명령에는 다음 매개변수가 포함됩니다.

List

선택적으로, 현재 로그 설정이 표시되도록 지정합니다. 명령 매개변수가 지정되지 않은 경우 기본값은 *List*입니다.

log_setting

선택적으로, 일련의 '0' 및 '1' 문자로 지정된 원하는 로그 설정을 나타냅니다.

- 0은 원하는 위치 로그 설정이 *OFF* 또는 사용 안함으로 설정됨을 나타냅니다.
- 1은 원하는 위치 로그 설정이 *ON* 또는 사용으로 설정됨을 나타냅니다.

지정된 값이 완전히 바뀌고, 현재 로그 설정과 병합되지 않습니다. 위치 로그 설정은 다음과 같습니다.

- 1 - ACSLS XAPI 구성 요소 로그에 오류 메시지를 기록합니다.
- 01 - 로그 파일에 메시지를 기록합니다.
- 001 - 로그 파일에 입력 트랜잭션 오류를 기록합니다.
- 0001 - 로그 파일에 모든 XML *recv()* 패킷을 기록합니다.
- 00001 - 로그 파일에 모든 XML *send()* 패킷을 기록합니다.
- 000001 - 로그 파일에 로컬 명령 및 응답을 기록합니다.

XAPI *LOG* 메시지는 `$ACS_HOME/log/xapi/vlog.file`에 저장됩니다.

주:

LOG 명령은 현재 위의 *LOG* 설정으로 정의된 0과 1의 여러 위치(16)를 설정 및 표시할 수 있게 해줍니다. 이러한 구성은 추가 확장을 허용하며, 여분의 0 또는 1 *LOG* 설정은 단순히 무시됩니다.

OFF

선택적으로, 해당 로깅이 사용 안함으로 설정되도록 지정합니다. 이 설정은 *LOG 0*과 동일합니다.

예

다음 예에서 사용자는 *LOG* 명령을 실행해서 로그 파일에 모든 오류 및 모든 XML *recv()* 패킷을 기록하도록 XAPI 서버를 설정합니다.

XCMD LOG 1011

LIST

다음 절에서는 *LIST* 명령에 대해 설명합니다.

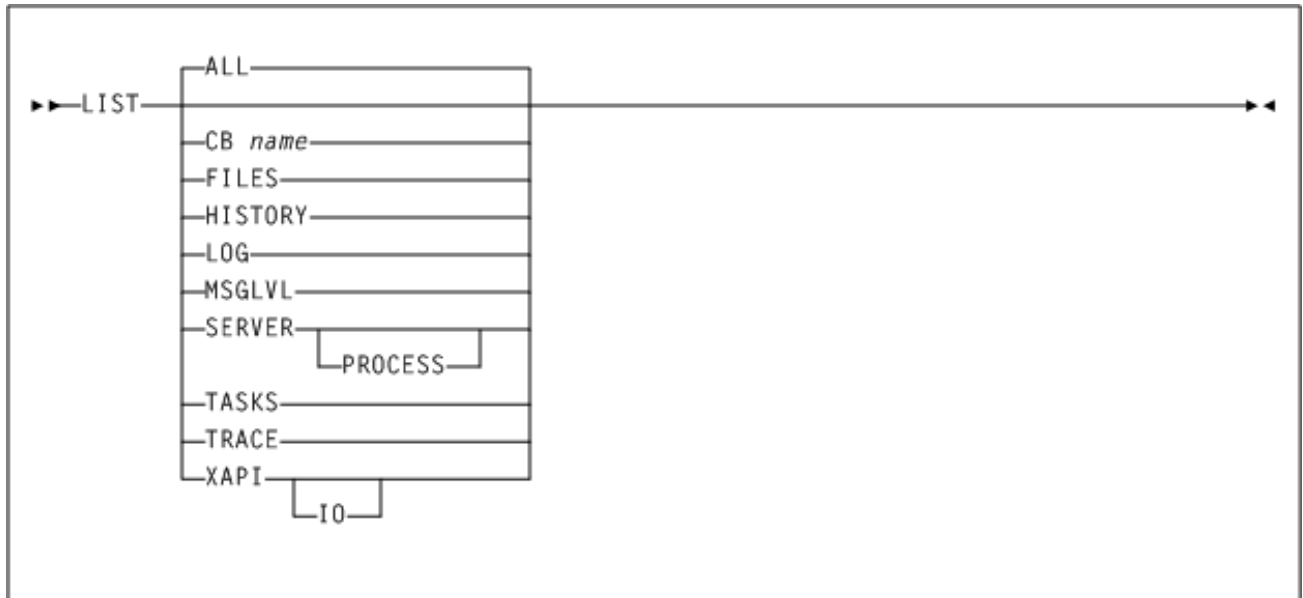
설명

LIST 명령은 현재 XAPI 서버 상태 및 설정을 나열합니다.

구문

다음 그림은 *LIST* 명령의 구문을 보여줍니다.

그림 4.2. *LIST* 명령 구문



매개변수

그림 4.2. “*LIST* 명령 구문”에 표시된 것처럼 *LIST* 명령에는 다음 매개변수가 포함됩니다.

ALL

선택적으로, 모든 XAPI 서버 상태 및 설정을 표시합니다. 이 값은 매개변수 없이 *LIST* 명령을 실행할 경우의 기본값입니다.

CB 이름

이름 제어 블록이 문자로 표시되도록 지정합니다. 문자는 16진수 형식입니다. 오라클 고객지원센터의 안내에 따라서만 사용하십시오.

*name*에 대해 다음 제어 블록 중 하나를 지정합니다.

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

LIST CB HTTPREQ 및 *LIST CB HTTPAPI*의 경우 0~999 사이의 인덱스를 지정해야 합니다. 또한, 인덱스는 중간 공백 없이 단일 대시("-")를 사용해서 제어 블록 이름과 구분되어야 합니다. 예를 들면 다음과 같습니다.

LIST HTTPREQ-0

또는

LIST HTTPAPI-900

FILES

선택적으로, XAPI 서버 제어, 로그 및 추적 파일의 전체 경로 이름이 표시되도록 지정합니다.

HISTORY

선택적으로, 이전 24시간 동안 XAPI 트랜잭션 수 내역이 표시되도록 지정합니다.

LOG

선택적으로, 현재 XAPI 서버 로그 설정이 표시되도록 지정합니다. 이 설정은 매개변수 없이 *LOG* 명령을 실행하는 것과 동일합니다.

MSGLVL

선택적으로, 현재 XAPI 서버 메시지 레벨 설정이 표시되도록 지정합니다. 이 설정은 매개변수 없이 *MSGLVL* 명령을 실행하는 것과 동일합니다.

SERVER

선택적으로, 현재 XAPI 서버 및 UNIX 버전, 관련 매개변수 및 환경 변수, 공유 세그먼트 및 메시지 대기열이 표시되도록 지정합니다.

선택적으로 *PROCESS* 키워드를 포함하여 모든 활성 XAPI 서버 프로세스 ID, 스레드 및 열린 파일 수, CPU 및 메모리 사용량도 표시되도록 요청할 수 있습니다.

PROCESS

*SERVER*로 지정된 경우, *PROCESS* 키워드는 모든 활성 XAPI 서버 프로세스 ID, 스레드 및 열린 파일 수, CPU 및 메모리 사용량도 표시되도록 요청합니다.

TASKS

선택적으로, 현재 XAPI 서버 시스템 및 작업이 표시되도록 지정합니다.

TRACE

선택적으로, 현재 XAPI 서버 추적 설정이 표시되도록 지정합니다. 이 설정은 매개변수 없이 *TRACE* 명령을 실행하는 것과 동일합니다.

XAPI

선택적으로 현재 XAPI 서버 리스너 포트 및 IP 주소가 표시되도록 지정합니다.

선택적으로 *IO* 키워드를 포함하여 모든 XAPI 서버 리스너 통계도 표시되도록 요청할 수 있습니다.

예

다음 예에서 사용자는 *List* 명령을 실행해서 현재 XAPI 서버 리스너 포트, IP 주소 및 리스너 상태를 표시합니다.

```
XCMD LIST XAPI IO
```

MSGLVL

다음 절에서는 *MSGLVL* 명령에 대해 설명합니다.

설명

MSGLVL 명령은 XAPI 서버의 메시지 상세 정보 표시를 변경 또는 나열합니다.

구문

다음 그림은 *MSGLVL* 명령의 구문을 보여줍니다.

그림 4.3. *MSGLVL* 명령 구문



매개변수

그림 4.3. “*MSGLVL* 명령 구문”에 표시된 것처럼 *MSGLVL* 명령에는 다음 매개변수가 포함됩니다.

List

현재 *MSGLVL* 설정이 표시되도록 지정합니다. 명령 매개변수가 지정되지 않은 경우 기본 값은 List입니다.

level

원하는 *MSGLVL* 설정을 지정합니다. 레벨이 높을수록 더 자세한 XAPI 서버 메시지가 표시됩니다. 메시지 레벨은 누적됩니다. *MSGLVL 8*은 최대 *MSGLVL 8*까지의 모든 메시지를 생성합니다. 즉, *MSGLVL 0~8*이 생성됩니다. 메시지 레벨은 다음과 같이 일반화됩니다.

- 0 - 일반 시작, 종료 및 오류 메시지: 이러한 메시지는 표시하지 않을 수 없습니다.
- 4 - 심각한 경고 메시지입니다.
- 8 - 사소한 경고 메시지입니다.
- 12 - 시작 매개변수 및 옵션 메시지입니다.

- 16 - 추가 시작 및 종료 메시지입니다.
- 20 - 추가 진단 레벨 20 메시지입니다.
- 24 - 추가 진단 레벨 24 메시지입니다.
- 28 - 추가 작업 시작 및 종료 메시지입니다.

OFF

모든 상세 정보 메시지를 사용 안함으로 설정하도록 지정합니다. 이 설정은 *MSGLVL 0*과 동일합니다.

예

다음 예에서 사용자는 *MSGLVL* 명령을 실행해서 모든 경고 메시지를 포함하도록 XAPI 서버 메시징을 변경합니다.

```
XCMD MSGLVL 8
```

TRACE

다음 절에서는 *TRACE* 명령에 대해 설명합니다.

설명

TRACE 명령은 XAPI 서버 추적 설정을 변경 또는 나열합니다.

주:

- 추적 파일 이름 및 위치는 ACSLS 환경 변수로 설정되며 기본적으로 *DV_TAG_XAPI_WORK_PATH* 변수에서 파일 이름 *vtrace.file*로 지정됩니다.
- 추적은 시스템 성능에 상당한 영향을 줄 수 있습니다. Oracle StorageTek 고객지원센터의 요청이 있을 때만 추적을 ON으로 설정하십시오.
- *TRACE* 명령은 현재 위의 *TRACE* 설정으로 정의된 것보다 많은 0과 1의 여러 위치(16)를 설정 및 표시할 수 있게 해줍니다. 이러한 구성은 추가 확장을 허용하며, 여분의 0 또는 1 *TRACE* 설정은 단순히 무시됩니다.

구문

다음 그림은 *TRACE* 명령의 구문을 보여줍니다.

그림 4.4. TRACE 명령 구문



매개변수

그림 4.4. “TRACE 명령 구문”에 표시된 것처럼 TRACE 명령에는 다음 매개변수가 포함됩니다.

List

선택적으로, 현재 추적 설정이 표시되도록 지정합니다. 명령 매개변수가 지정되지 않은 경우 기본값은 *List*입니다.

trace_setting

선택적으로 일련의 '0' 및 '1' 문자로 지정된 원하는 추적 설정을 나타냅니다. 0은 원하는 위치 추적 설정이 *OFF* 또는 사용 안함으로 설정됨을 나타내고, 1은 원하는 위치 추적 설정이 *ON* 또는 사용으로 설정됨을 나타냅니다. 지정된 값이 완전히 바뀌고, 현재 추적 설정과 병합되지 않습니다. 위치 추적 설정은 다음과 같습니다.

- 1 - 추적 파일로 오류를 추적합니다.
- 01 - 추적 파일로 TCP/IP 기능 및 이벤트를 추적합니다.
- 001 - 추적 파일로 PGMI 또는 ACSAPI 기능 및 이벤트를 추적합니다.
- 0001 - 추적 파일로 분류되지 않은 XAPI 서버 이벤트를 추적합니다.
- 00001 - 추적 파일로 malloc() 및 free() 이벤트를 추적합니다.
- 000001 - 추적 파일로 XML 구문 분석기 이벤트를 추적합니다.
- 0000001 - 추적 파일로 명령 서버 프로세스 이벤트를 추적합니다.
- 00000001 - 추적 파일로 모니터 프로세스 이벤트를 추적합니다.
- 000000001 - 추적 파일로 CSV 기능 및 이벤트를 추적합니다.

OFF

선택적으로, 추적이 사용 안함으로 설정되도록 지정합니다. 이 설정은 *TRACE 0*과 동일합니다.

예

다음 예에서 사용자는 *TRACE* 명령을 실행해서 모든 오류, 모든 malloc() 및 free() 이벤트를 추적 파일로 추적하도록 XAPI 서버를 설정합니다.

```
XCMD TRACE 10001
```

5장. XAPI 사용자 요청 및 HSC/VTCS 명령에 대한 XCMD 지원

3장. XCMD XAPI 구성 및 관리 명령 에 설명된 XAPI 서버 구성 및 관리 명령 및 4장. XCMD XAPI 서버 운영자 명령 에 설명된 XAPI 서버 운영자 명령 외에도 XCMD 인터페이스 는 여러 XAPI 사용자 요청 및 호환 가능한 HSC/MVS 텍스트 명령을 지원합니다.

예를 들어, XCMD 명령을 사용해서 ACSLS user_proc에서 QUERY_VOLUME 요청을 입력하려면 다음을 수행합니다.

```
XCMD QUERY_VOLSER S00001
```

지원되는 XAPI 요청, 명령 및 매개변수

다음 표에는 XCMD에서 지원되는 XAPI 요청 및 해당 HSC/MVS 텍스트 명령이 나열되어 있습니다.

표 5.1. XCMD 지원 XAPI 요청 및 해당 HSC/MVS 명령

XAPI 요청	해당 HSC/MVS 텍스트 명령
DISMOUNT	DISMount
EJECT	EJect
MOUNT	Mount
MOVE	MOVE
QUERY_CAP	Display Cap
QUERY_DRIVES	Display DRives
QUERY_DRIVE_INFO	Display DRIVE_INFO
QUERY_LSM	Display Lsm
QUERY_SERVER	해당하지 않는 HSC/MVS 텍스트 명령
QUERY_SCRATCH	Display SCRatch
QUERY_THRESHOLD	Display THReshold
QUERY_VOLSER 또는 QUERY_VOLUME	Display Volser
QUERY_VOLUME_INFO	Display VOLume_info
SCRATCH	SCRatch
UNSCRATCH	UNSCRatch

XAPI 요청	해당 HSC/MVS 텍스트 명령
<i>VOLRPT</i> ¹	<i>VOLRpt</i>

¹*VOLRPT*는 HSC 형식 *VOLRPT* 명령 및 보고서이며, ACSLS 형식 *volrpt*가 아닙니다.

위에 나열된 HSC/MVS 텍스트 명령에 대한 자세한 내용은 *ELS Command, Control Statement, and Utility Reference*를 참조하십시오.

이러한 요청 및 명령이 지원되더라도 ACSLS에서 모든 매개변수가 유효한 것은 아닙니다. 잘못된 매개변수는 다음과 같습니다.

- *DISMOUNT*(또는 *DISMount* 명령)

다음 항목을 지정하면 구문 오류가 발생합니다.

- MVS 장치 주소 형식 *ccuu*
- *hostid* 위치 매개변수

- *EJECT*(또는 *EJect* 명령)

다음 매개변수를 지정하면 구문 오류가 발생합니다.

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

- *MOUNT*(또는 *Mount* 명령)

다음 항목을 지정하면 구문 오류가 발생합니다.

- MVS 장치 주소 형식 *ccuu*
- *hostid* 위치 매개변수
- *MGMTCLAS* 매개변수는 잘못되었습니다.

- *MOVE*(또는 *MOVE* 명령)

모든 매개변수가 지원됩니다.

- *QUERY_CAP*(또는 *Display Cap* 명령)

- 모든 매개변수가 지원됩니다.

- *QUERY_DRIVES*(또는 *Display DRives* 명령)

다음 매개변수를 지정하면 구문 오류가 발생합니다.

- *ALL*
- *IDLE*
- *LIBRARY*
- *DETAIL*
- *IDENTITY*
- *MEDIA*
- *RECTECH*

- *UNIT*
- *BYDRIVE*
- *BYLOC*
- *SHOWLSLOT*
- *QUERY_DRIVE_INFO*(또는 *Display DRIVE_INFO* 명령)

다음 매개변수를 지정하면 구문 오류가 발생합니다.

 - *ALL*
 - *IDLE*
 - *LIBRARY*
 - *DETAIL*
 - *IDENTITY*
 - *MEDIA*
 - *RECTECH*
 - *VIRTUAL*
 - *DEVADDR*
 - *LSMLOC*
- *QUERY_LSM*(또는 *Display Lsm* 명령)

모든 매개변수가 지원됩니다.
- *QUERY_SERVER*(또는 *Display SERVER* 명령)

모든 매개변수가 지원됩니다.
- *QUERY_SCRATCH*(또는 *Display SCRatch* 명령)

모든 매개변수가 지원됩니다.
- *QUERY_THRESHOLD*(또는 *Display THReshold* 명령)

모든 매개변수가 지원됩니다.
- *QUERY_VOLSER*(또는 *Display Volser* 명령)

모든 매개변수가 지원됩니다.
- *QUERY_VOLUME_INFO*(또는 *Display VOLUME_info* 명령)

모든 매개변수가 지원됩니다.
- *SCRATCH*(또는 *SCRatch* 명령)

모든 매개변수가 지원됩니다.
- *UNSCRATCH*(또는 *UNSCRatch* 명령)

모든 매개변수가 지원됩니다.
- *VOLRPT*(또는 *VOLRpt* 명령)

다음 항목을 지정하면 구문 오류가 발생합니다.

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

SMC 또는 VM 클라이언트 경로 지정 명령을 사용해서 ACSLS XAPI 서버에 대한 명령 실행

EJECT 또는 *VOLRpt* 명령을 제외하고 *Route* 명령을 사용하여 ACSLS XAPI 서버에 대해 SMC/MVS 클라이언트 또는 VM/클라이언트에서 표 5.1. “XCMD 지원 XAPI 요청 및 해당 HSC/MVS 명령”에 나열된 HSC/MVS 명령을 실행할 수 있습니다.

ACSLs cmd_proc에서 입력한 *XCMD* 명령과 달리 SMC/MVS 또는 VM/클라이언트 *Route* 명령은 설정된 SMC/HSC 구문 규약을 적용합니다. 또한 *XCMD* 토큰은 생략됩니다.

예를 들면 다음과 같습니다.

XAPI *QUERY VOLUME* 요청을 ACSLS cmd_proc *XCMD* 명령으로 실행하려면 다음을 수행합니다.

```
XCMD QUERY VOLUME VOLSER
```

여기서 *VOLSER*는 원하는 볼륨입니다.

ROUTE 명령을 사용해서 ACSLS XAPI 서버에 대해 SMC/MVS 클라이언트에서 XAPI *QUERY VOLUME* 요청을 실행하려면 다음을 수행합니다.

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

여기서 *vo1ser*는 원하는 볼륨입니다.

SMC *ROUTE* 명령에 대한 자세한 내용은 *ELS Command, Control Statement, and Utility Reference*를 참조하십시오.

6장. ACSLS XAPI 서버 제어 변수

Oracle ACSLS 소프트웨어는 ACSLS 시스템의 특정 동작을 제어할 수 있게 해주는 시스템 변수 세트를 제공합니다. ACSLS XAPI 서버 구성 요소와 관련된 일부 제어 변수가 포함됩니다. 이러한 변수에는 다음이 포함됩니다.

- 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

ACSLX XAPI 서버 제어 변수 설정

ACSLX *acsxs_config* 또는 *dv_config* 유틸리티를 사용해서 ACSLS XAPI 정적 변수를 표시 및 설정합니다. 변경사항을 적용하려면 ACSLS를 다시 시작해야 합니다.

- *dv_config* 유틸리티를 사용하려면 다음 명령 중 하나를 입력합니다.
 - *dv_config -d* - 모든 ACSLS 변수를 표시합니다.
 - *dv_config -p <variable_name>* - XAPI 변수를 업데이트합니다.
- *acsxs_config* 유틸리티를 사용하려면 ACSLS *acsxs_config* 스크립트를 사용해서 ACSLS 기능 구성 화면에 액세스합니다.

ACSLX 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

옵션 9(*Set XAPI server variables*)를 선택합니다.

주:

이 옵션은 ACSLS XAPI 서버 기능을 사용으로 설정한 경우에만 표시됩니다.

acsss_config 및 *dv_config* 유틸리티와 이러한 유틸리티를 사용해서 ACSLS 변수를 표시 및 업데이트하는 방법에 대한 자세한 내용은 *StorageTek Automated Cartridge System Library Software Administrator's Guide*를 참조하십시오.

ACSLX XAPI 변수 설명

이 절에서는 ACSLS XAPI 서버 구성 요소를 제어하기 위해 사용되는 유효한 변수에 대해 설명합니다.

XAPI_PORT

프롬프트: XAPI 서버에 대한 사용자 정의 인바운드 포트 변경사항은 XAPI 서버를 다시 시작해야 적용됩니다. 수신 XAPI 요청을 받기 위해 XAPI 서버에서 사용되는 포트 번호입니다. [50020]:

이 옵션은 클라이언트의 TCP 요청 수신을 위해 XAPI 서버에서 사용되는 포트를 지정합니다. XAPI 서버에서 사용되는 포트를 정의하려면 1024에서 65535 사이의 숫자를 입력하십시오. 포트 50003은 지정하지 마십시오. 자세한 내용은 ACSLS 보안 설명서를 참조하십시오.

이 변수를 적용하려면 XAPI 서버를 다시 시작해야 합니다.

XAPI_WORK_PATH

프롬프트: XAPI 작업 디렉토리 변경사항은 XAPI 서버를 다시 시작해야 적용됩니다. XAPI 로그 및 추적 정보를 [*\$ACS_HOME/log/xapi*] 디렉토리에 배치합니다.

XAPI 서버 작업 파일이 배치되는 디렉토리를 선택합니다. 설치되면 XAPI 서버가 정보를 *\$ACS_HOME/log/xapi* 디렉토리에 기록합니다. 일반적인 사용에서 이 변수의 값은 변경되지 않습니다. *\$ACS_HOME*이 포함된 파일 시스템에 디스크 공간 문제가 있는 경우 대체 경로를 지정할 수 있습니다. 제공된 경로는 절대 경로여야 합니다(예: / 또는 *\$ACS_HOME*으로 시작하는 경로). 이 변수를 적용하려면 XAPI 서버를 다시 시작해야 합니다.

XAPI_LOG_SIZE

프롬프트: XAPI 로그 크기 변경사항은 XAPI 서버를 다시 시작해야 적용됩니다. 최대 XAPI 로그 크기(MB)입니다. [20]:

이 옵션에서는 MB로 표현된 XAPI 로그에 대한 임계값 크기를 지정합니다(여기서는 "1048576바이트"로 정의됨). 음수가 아닌 숫자를 입력합니다. 이 옵션의 기본값은 20입니다.

이 변수를 적용하려면 XAPI 서버를 다시 시작해야 합니다.

XAPI_LOG_FILE_NUM

프롬프트: 보존할 XAPI 로그 아카이브 파일 수입니다. [10]:

이 옵션에서는 보존할 아카이브된 XAPI 로그 파일 수를 지정합니다.

현재 vlog.file 크기가 임계값 크기를 초과할 경우 로그 파일의 n 접미어에 0이 추가되어 이름이 바뀝니다. 0은 최신 파일이고 n 은 가장 오래된 파일입니다. 아카이브된 파일은 `XAPI_WORK_PATH` 디렉토리에 저장됩니다.

지정된 아카이브된 로그 수에 도달하는 경우 새 파일이 아카이브 디렉토리에 추가될 때마다 가장 오래된 파일이 이 디렉토리에 제거됩니다. 1개에서 99개 사이의 아카이브된 파일을 보존할 수 있습니다.

보존할 아카이브된 로그 파일 수를 지정하려면 1에서 99 사이의 숫자를 입력합니다.

이 변수를 적용하려면 XAPI 서버를 다시 시작해야 합니다.

XAPI_TRACE_SIZE

프롬프트: XAPI 추적 크기 변경사항은 xapi 서버를 다시 시작해야 적용됩니다. 최대 XAPI 추적 크기(MB)입니다. [50]:

이 옵션은 XAPI 추적에 대한 임계값 크기(MB)를 지정합니다(여기서는 "1048576바이트"로 정의됨). 음수가 아닌 숫자를 입력합니다. 이 옵션의 기본값은 50입니다.

이 변수를 적용하려면 XAPI 서버를 다시 시작해야 합니다.

XAPI_TRACE_FILE_NUM

프롬프트: 보존할 XAPI 추적 아카이브 파일 수입니다. [10]:

이 옵션은 보존할 아카이브된 XAPI TRACE 파일 수를 지정합니다. 현재 vtrace.file 크기가 임계값 크기를 초과할 경우 추적 파일의 n 접미어에 0이 추가되어 이름이 바뀝니다. 0은 최신 파일이고 n 은 가장 오래된 파일입니다. 아카이브된 파일은 `XAPI_WORK_PATH` 디렉토리에 저장됩니다. 지정된 아카이브된 로그 수에 도달하는 경우 새 파일이 아카이브 디렉토리에 추

가 될 때마다 가장 오래된 파일이 이 디렉토리에서 제거됩니다. 아카이브 파일은 1~99개까지 보존할 수 있습니다. 보존할 아카이브된 로그 파일 수를 지정하려면 1에서 99 사이의 숫자를 입력합니다.

이 변수를 적용하려면 XAPI 서버를 다시 시작해야 합니다.

XAPI_STARTUP_FILE

프롬프트: XAPI 시작 파일 이름 변경사항은 XAPI 서버를 다시 시작해야 적용됩니다. 제어 매개변수 [*xapi_startup_file*]이 포함된 XAPI 시작 파일의 이름

이 옵션은 XPI 시작 파일의 이름을 지정합니다. 이 파일은 *XAPI_WORK_PATH* 디렉토리에 있으며 XAPI 시작 매개변수를 포함합니다.

이 변수를 적용하려면 XAPI 서버를 다시 시작해야 합니다.

XAPI_TAPEPLEX_NAME

프롬프트: XAPI Tapeplex 이름 변경사항은 XAPI 서버를 다시 시작해야 적용됩니다. XAPI Tapeplex의 이름 []:

이 옵션에서는 XAPI Tapeplex의 이름을 지정합니다. 길이가 8자를 초과하지 않는 텍스트를 입력하십시오. 이 변수를 적용하려면 XAPI 서버를 다시 시작해야 합니다.

색인

기호

ACSLX XAPI 제어 변수, 41
DEFINE POOL_Access 명령, 20
DEFINE POOL_Name 명령, 19
DELETE POOL_Access 명령, 23
DELETE POOL_Name 명령, 22
LIST 명령, 31
LOG 명령, 29
MSGLVL 명령, 33
QUERY POOL_Access 명령, 27
QUERY POOL_Name 명령, 26
SET POOL_Name 명령, 24
TRACE 명령, 34

ㄱ

관리 명령, 19
구문

DEFINE POOL_Access 명령, 20
DEFINE POOL_Name 명령, 19
DELETE POOL_Access 명령, 23
DELETE POOL_Name 명령, 22
LOG 명령, 29, 31
MSGLVL 명령, 33
QUERY POOL_Access 명령, 28
QUERY POOL_Name 명령, 26
SET POOL_Name 명령, 24
TRACE 명령, 34

구성 명령, 19
규약, 명령, 17

ㄴ

명령

DEFINE POOL_Name, 19
DEFINE POOL_Access, 20
DELETE POOL_Name, 22
DELETE POOL_Access, 23
LIST, 31
LOG, 29
MSGLVL, 33
QUERY POOL_Name, 26
QUERY POOL_Access, 27
SET POOL_Name, 24

TRACE, 34

XAPI 구성 및 관리 명령, 19

XAPI 서버 운영자 명령, 29

명령 규약, 17

명령 입력

ACSLX cmd-proc에서, 17

SMC 또는 VM 클라이언트 경로 지정 명령 사용,
40

ㄷ

변수, ACSLS, 41

ㄹ

설명

ACSLX 서버에 대한 XAPI 클라이언트 인터페이스,
15

XCMD 인터페이스 구성 요소, 15

소개, 15

ㅇ

예

DEFINE POOL_Access, 20, 22
DELETE POOL_Access 명령, 24
DELETE POOL_Name 명령, 23
LOG 명령, 30, 33
MSGLVL 명령, 34
QUERY POOL_Access 명령, 28
QUERY POOL_Name 명령, 27
Route 명령, 40
SET POOL_Name, 26
TRACE 명령, 35

운영자 명령, 29
