

HP-UX 시스템 관리 설명서: 논리 볼륨 관리

HP-UX 11i v3

HP 제품 번호: 5992-4592
2008년 9월 발행
제3판



알림

기밀 컴퓨터 소프트웨어입니다. 소유, 사용 또는 복사를 위해서는 HP로부터 유효한 라이선스를 취득해야 합니다. FAR 12.211 및 12.212에 따라 상업용 컴퓨터 소프트웨어, 컴퓨터 소프트웨어 문서 및 상업용 품목의 기술 데이터는 공급업체의 표준 상업용 라이선스에 의거하여 미국 정부에 사용이 허가되었습니다. 이 설명서의 내용은 예고 없이 변경될 수 있습니다. HP 제품과 서비스에 대한 보증은 오직 제품 및 서비스와 함께 제공되는 명시적 보증서만을 근거로 합니다. 이 설명서의 어떤 내용도 추가 보증 제정으로 해석할 수 없습니다. HP는 이 문서에 포함된 기술적 오류나 편집상의 오류에 대해 책임을 지지 않습니다. UNIX는 The Open Group의 등록 상표입니다.

상표권

VERITAS™는 Symantec Corporation의 등록 상표입니다.

목차

설명서 정보.....	11
대상 독자.....	11
이 설명서의 새로운 내용 및 변경된 내용.....	11
시리즈 정보.....	11
설명서 구성.....	11
표기법.....	12
예제 및 셀.....	13
명령 구문.....	13
관련 정보.....	14
HP-UX 정보 찾기.....	14
HP-UX 11i 릴리즈 이름 및 운영 체제 버전 ID.....	15
시스템 버전 확인.....	16
발행 정보.....	17
사용자 의견 접수.....	17
 1 소개.....	 19
LVM 기능.....	19
LVM 아키텍처.....	20
물리 확장 영역과 논리 확장 영역 비교.....	21
LVM 볼륨 그룹 버전.....	23
LVM 장치 파일 사용.....	24
기존 장치 파일과 영구 장치 파일 비교.....	25
LVM의 명명 규칙.....	25
장치 번호 형식.....	27
버전 1.0 장치 번호 형식.....	28
버전 2.x 장치 번호 형식.....	28
LVM 디스크 레이아웃.....	28
Boot Data Reserved Area.....	29
Logical Interface Format Area.....	29
Physical Volume Reserved Area.....	29
Volume Group Reserved Area.....	30
사용자 데이터 영역.....	30
LVM 제한 사항.....	30
 2 LVM 구성.....	 33
LVM 구성 계획.....	33
여러 유형의 논리 볼륨 설정.....	33
원시 데이터 저장소용 논리 볼륨 설정.....	33
파일 시스템용 논리 볼륨 설정.....	34

파일 시스템 논리 볼륨 지칭.....	36
스왑용 논리 볼륨 설정.....	36
스왑 논리 볼륨 지칭.....	36
덤프용 논리 볼륨 설정.....	37
덤프 논리 볼륨 지칭.....	37
가용성 계획.....	37
미러링을 통해 데이터 가용성 늘리기.....	38
미러 쓰기 동작 제어.....	38
할당 정책.....	38
예약 정책.....	39
동기화 정책.....	40
미러 논리 볼륨 동기화.....	40
디스크 스페어링을 통해 디스크 중복 늘리기.....	42
다중 경로 지정을 통해 하드웨어 경로 중복 늘리기.....	43
물리 볼륨에 대한 다중 경로 지정 설정.....	43
성능 계획.....	44
일반 성능 요소.....	44
메모리 사용.....	44
CPU 사용.....	44
디스크 공간 사용.....	45
내부 성능 요소.....	45
예약 정책.....	45
미러 쓰기 일관성 캐시.....	45
디스크 스패닝.....	46
볼륨 그룹의 수.....	46
물리 볼륨 그룹.....	46
디스크 스트라이핑을 통한 성능 향상.....	46
최적 스트라이프 크기 결정.....	48
미러링과 스트라이핑 사이의 상호 작용.....	48
I/O 채널 분리를 통한 성능 향상.....	48
복구 계획.....	49
LVM 시스템 복구 준비.....	50
LVM 구성 기록의 예제 스크립트.....	51
3 LVM 관리.....	53
관리 도구.....	53
LVM 정보 표시.....	56
볼륨 그룹 정보.....	56
물리 볼륨 정보.....	57
논리 볼륨 정보.....	57
일반적인 LVM 작업.....	58
LVM 사용을 위한 디스크 초기화.....	59
볼륨 그룹 만들기.....	60

볼륨 그룹 장치 파일 만들기.....	60
버전 1.0 볼륨 그룹 만들기.....	60
버전 2.x 볼륨 그룹 만들기.....	61
볼륨 그룹에 디스크 추가.....	62
볼륨 그룹에서 디스크 제거.....	62
논리 볼륨 만들기.....	63
스트라이프 논리 볼륨 만들기.....	64
미러 논리 볼륨 만들기.....	64
논리 볼륨 확장.....	65
논리 볼륨을 특정 디스크로 확장.....	66
논리 볼륨 축소.....	66
논리 볼륨에 미러 추가.....	67
논리 볼륨에서 미러 제거.....	68
논리 볼륨 이름 바꾸기.....	68
논리 볼륨 제거.....	69
볼륨 그룹 내보내기.....	69
볼륨 그룹 가져오기.....	70
볼륨 그룹 매개 변수 수정.....	70
볼륨 그룹 정지 및 다시 시작.....	75
볼륨 그룹 이름 바꾸기.....	76
볼륨 그룹 분리.....	77
볼륨 그룹 제거.....	78
미러 논리 볼륨 백업.....	79
볼륨 그룹 구성 백업 및 복원.....	80
디스크 이동 및 재구성.....	81
시스템 내에서 디스크 이동.....	82
시스템 간 디스크 이동.....	83
다른 물리 볼륨으로 데이터 이동.....	84
스페어 디스크 만들기.....	86
스페어 디스크 복귀.....	86
물리 볼륨 특성 수정.....	87
크기 변경 인식.....	87
물리 볼륨 부팅 유형 변경.....	91
물리 볼륨에 대한 경로 비활성화.....	94
대체 부팅 디스크 만들기.....	95
부팅 디스크 미러링.....	99
HP 9000 서버에서 부팅 디스크 미러링.....	99
HP Integrity 서버에서 부팅 디스크 미러링.....	101
파일 시스템 논리 볼륨 관리.....	105
파일 시스템 만들기.....	105
파일 시스템 확장.....	106
파일 시스템 크기 축소.....	108
VxFS 스냅샷 파일 시스템 백업.....	110
스왑 논리 볼륨 관리.....	111

스왑 논리 볼륨 만들기.....	111
스왑 장치 확장.....	111
스왑 장치 크기 축소.....	112
덤프 논리 볼륨 관리.....	112
덤프 논리 볼륨 만들기.....	112
덤프 논리 볼륨 제거.....	112
하드웨어 문제.....	112
복제된 LUN 통합.....	113
4 LVM 문제 해결.....	115
문제 해결 도구 개요.....	115
정보 모음.....	115
일관성 검사.....	116
유지 관리 모드 부팅.....	116
로그 파일 및 추적 파일.....	117
I/O 오류.....	117
복구 가능한 오류.....	118
일시적으로 사용할 수 없는 장치.....	118
영구적으로 사용할 수 없는 장치.....	118
복구 불가능 오류.....	119
미디어 오류.....	119
볼륨 그룹 활성화 시 장치 누락.....	120
볼륨 그룹 활성화 장애.....	120
비 루트 볼륨 그룹의 퀴럼 문제.....	121
루트 볼륨 그룹의 퀴럼 문제.....	121
버전 2.x 볼륨 그룹 활성화 장애.....	122
루트 볼륨 그룹 검색.....	123
LVM 부팅 장애.....	124
퀴럼 부족.....	124
디스크의 손상된 LVM 데이터 구조.....	124
손상된 LVM 구성 파일.....	125
논리 볼륨의 크기를 축소시킨 후 발생하는 문제.....	125
손상된 디스크 교체.....	126
디스크 교체 전제 조건.....	126
미려된 부팅할 수 없는 디스크 교체.....	128
미려되지 않은 부팅할 수 없는 디스크 교체.....	131
미려 부팅 디스크 교체.....	135
미려되지 않은 부팅 디스크 교체.....	138
경고 및 오류 메시지.....	138
물리 디스크 및 볼륨 그룹과 오류 메시지 일치.....	138
모든 LVM 명령에 대한 메시지.....	140
lvchange(1M).....	140
lvextend(1M).....	140

lvlnboot(1M).....	142
pvchange(1M).....	142
vgcfgbackup(1M).....	143
vgcfgrestore(1M).....	143
vgchange(1M).....	144
vgcreate(1M).....	146
vgdisplay(1M).....	147
vgextend(1M).....	148
vgimport(1M).....	149
/var/adm/syslog/syslog.log.....	149
문제 보고.....	151
 A LVM 사양 및 제한.....	153
 B LVM 명령 요약.....	157
 C 볼륨 그룹 프로비저닝 팁.....	161
버전 1.0 볼륨 그룹의 최적 확장 영역 크기 선택.....	161
 용어.....	163
 색인.....	165

그림 목록

1-1	논리 볼륨으로 파티션된 디스크 공간.....	21
1-2	물리 확장 영역 및 논리 확장 영역.....	22
2-1	파일 시스템 공간 구성 요소.....	34
2-2	버스 사이에 디스크 인터리빙.....	47
3-1	HP Integrity 서버의 LVM 디스크 레이아웃 예제.....	102

표 목록

1	HP-UX 정보 찾기.....	15
2	HP-UX 11i 릴리즈.....	16
3	OS 버전, 시스템 아키텍처 및 시스템 모델.....	16
1-1	LVM 볼륨 그룹 버전 등록 정보.....	24
1-2	물리 볼륨 명명 규칙.....	26
1-3	버전 1.0 장치 번호 형식.....	28
1-4	버전 2.x 장치 번호 형식.....	28
3-1	물리 볼륨 관리 명령.....	54
3-2	볼륨 그룹 관리 명령.....	54
3-3	논리 볼륨 관리 명령.....	55
4-1	수집 및 유지 관리할 LVM 정보.....	115
A-1	볼륨 그룹 버전 최대값.....	153
A-2	버전 1.0 볼륨 그룹 제한.....	154
A-3	버전 2.x 볼륨 그룹 제한.....	155
B-1	LVM 명령 요약.....	157

설명서 정보

HP-UX 시스템 관리 설명서: 논리 볼륨 관리에서는 HP-UX 11i v3 플랫폼에서 LVM(Logical Volume Manager) 제품을 구성하고 관리하며 문제를 해결하는 방법에 대해 설명합니다.

대상 독자

HP-UX 시스템 관리 설명서는 HP-UX 릴리즈 11i v3부터 HP-UX 시스템을 관리해야 하는 모든 기술 수준의 HP-UX 시스템 관리자를 위해 작성되었습니다.

이 설명서 세트의 많은 항목은 이전 릴리즈에 적용되지만 HP-UX 11i v3에서 변경된 사항도 많습니다. 따라서 이전 릴리즈에 대한 자세한 내용은 **시스템 및 작업 그룹 관리: HP-UX 시스템 관리자를 위한 설명서**를 참조하십시오.

이 설명서의 새로운 내용 및 변경된 내용

HP-UX 시스템 관리 설명서: 논리 볼륨 관리의 제3판에서는 다음과 같은 새로운 내용에 대해 설명합니다.

- HP-UX 11i v3의 2008년 9월 릴리즈에 도입된 버전 2.1 볼륨 그룹. 자세한 내용은 “LVM 볼륨 그룹 버전” (23 페이지)을 참조하십시오.
- 볼륨 그룹 프로비저닝 권장 사항. 자세한 내용은 **부록 C (161 페이지)**를 참조하십시오.

시리즈 정보

HP-UX 시스템 관리 설명서에서는 HP-UX 11i v3을 실행하는 시스템 관리에 필요한 핵심 작업 및 관련 개념에 대해 설명합니다.

HP-UX 시스템 관리 설명서는 다음 볼륨으로 구성된 설명서 세트입니다.

개요	HP-UX 11i 와 그 구성 요소, 이들 사이의 관계에 대해 자세히 설명합니다.
구성 관리	시스템 설정 및 하위 시스템의 동작을 구성 및 사용자 정의하기 위해 수행해야 할 작업 중 많은 부분에 대해 설명합니다.
논리 볼륨 관리	HP LVM(Logical Volume Manager)을 사용하여 물리 볼륨, 볼륨 그룹 및 논리 볼륨을 구성하는 방법에 대해 설명합니다.
보안 관리	HP-UX 11i의 데이터 및 시스템 보안 기능에 대해 설명합니다.
루틴 관리 작업	시스템이 원활하게 실행되도록 유지하기 위해 수행해야 할 지속적인 여러 작업에 대해 설명합니다.

설명서 구성

이 설명서는 다음과 같이 구성되어 있습니다.

1장 소개

이 장에서는 LVM 개념에 대해 설명합니다.

2장 LVM 구성

3장 LVM 관리

4장 LVM 문제 해결

부록 A LVM 사양 및 제한

부록 B LVM 명령 요약

부록 C 볼륨 그룹 프로비저닝 팁

용어

이 장에서는 LVM을 사용하도록 시스템을 구성하는 방법에 대해 설명합니다.

이 장에서는 지속적으로 LVM을 관리하는 방법에 대해 설명합니다.

이 장에서는 LVM 구성에 있는 문제를 해결하는 방법에 대해 설명합니다.

이 부록에서는 제품 사양을 찾을 수 있습니다.

이 부록에서는 자주 사용하는 LVM 명령을 찾을 수 있습니다.

볼륨 그룹 작성 매개 변수에 대한 권장 사항이 찾아보려면 이 부록을 참조하십시오.

이 부록에서는 자주 사용하는 LVM 용어의 정의를 찾을 수 있습니다.

표기법

이 설명서에서 사용하는 표기법은 다음과 같습니다.

%, \$ 또는 #

백분율 기호는 C 셸 시스템 프롬프트를 나타냅니다. 달러 기호는 Bourne, Korn 및 POSIX 셸의 시스템 프롬프트를 나타냅니다. 번호 기호는 슈퍼유저 프롬프트를 나타냅니다.

audit(5)

맨페이지입니다. 해당 맨페이지 이름은 **audit**이고 5절에 있습니다.

Command

명령 이름이거나 명령 구문입니다.

Computer output

컴퓨터 화면에 표시되는 텍스트입니다.

Ctrl+x

키 시퀀스입니다. **Ctrl+x**와 같은 시퀀스는 **Ctrl**이라고 표시된 키를 누른 채로 다른 키나 마우스 단추를 눌러야 하는 것을 나타냅니다.

Document Title

설명서의 제목입니다. 웹 및 Instant Information 미디어에서는 설명서 자체에 대한 링크일 수 있습니다.

ENVIRONMENT VARIABLE

PATH 등의 환경 변수 이름입니다.

[ERROR NAME]

오류 이름이며 대개 `errno` 변수로 반환됩니다.

Key

키보드 키 이름입니다. **Return** 키와 **Enter** 키는 같은 키를 나타냅니다.

Term

한정되어 사용하는 중요한 단어나 문장입니다.

User input	사용자가 입력하는 명령이나 텍스트입니다.
Variable	명령, 함수 또는 기타 구문 표시에서 실제 값으로 바뀌어야 할 자리 표시자의 이름입니다.
[]	구문에서 내용이 선택 사항인 것을 나타냅니다. 내용이 로 구분된 목록인 경우 그 중 하나를 선택해야 합니다.
{ }	구문에서 내용이 필수 사항인 것을 나타냅니다. 내용이 로 구분된 목록인 경우 그 중 하나를 선택해야 합니다.
...	앞에 있는 요소를 원하는 만큼 반복할 수 있습니다.
	선택 목록에서 항목을 구분합니다.
경고	경고는 잘 이해하고 따르지 않을 경우 부상이나 복구 불가능한 시스템 문제를 일으킬 수 있는 중요 정보에 대한 주의를 환기시킵니다.
주의	주의는 잘 이해하고 따르지 않을 경우 데이터 손실, 데이터 손상 또는 하드웨어나 소프트웨어 손상을 유발할 수 있는 중요 정보에 대한 주의를 환기시킵니다.
중요	이 경고는 개념을 설명하거나 작업을 완료하는 데 필요한 정보를 제공합니다.
참고	참고에는 본문의 중요한 내용을 강조하거나 보충하기 위한 추가 정보가 포함됩니다.

예제 및 셸

이 설명서에서는 시스템 관리자가 사용하는 방법에 대해 설명합니다. POSIX 셸 `/sbin/sh`를 사용하려면 `root` 사용자(수퍼유저)가 있어야 하므로 모든 명령 예제에서는 해당 셸을 사용합니다. POSIX 셸은 `sh-posix(1)`에 정의되어 있습니다. 다른 셸에 대한 자세한 내용은 *Shells User's Guide* 및 `sh(1)`를 참조하십시오.

명령 구문

Literal	실제로 입력하는 단어 또는 문자입니다.
Replaceable	알맞은 값으로 바꾸는 단어 또는 구입니다.
-chars	하나 이상의 그룹화된 명령 옵션입니다(예: <code>-ikx</code>). 일반적으로 <code>chars</code> 는 각각 특정 옵션을 나타내는 리터럴 문자로 구성된 문자열입니다. 예를 들어, <code>-ikx</code> 항목은 개별 옵션인 <code>-i</code> ,

	-k 및 -x와 같습니다. 더하기 문자(+)가 옵션 접두어로 사용되는 경우도 있습니다.
-word	단일 명령 옵션입니다(예: -help). word 는 리터럴 키워드입니다. -chars와의 차이가 분명한 편이며, 옵션 설명에서 명확하게 구분됩니다. 더하기 문자(+)와 이중 하이픈(--)은 옵션 접두어로 쓰이는 경우도 있습니다.
[]	대괄호 메타 문자는 선택적인 내용을 형식 및 명령 설명으로 묶습니다.
{ }	중괄호 메타 문자는 필수적인 내용을 형식 및 명령 설명으로 묶습니다.
	막대 메타 문자는 주로 대괄호나 중괄호로 묶인 선택 목록에서 선택 항목을 분리합니다.
...	토큰 다음에 오는 생략 부호 메타 문자(abc...) 또는 오른쪽 대괄호([]...)나 오른쪽 중괄호({ }...) 메타 문자는 앞의 요소와 그 앞에 오는 공백이 임의의 횟수만큼 반복될 수 있음을 나타냅니다.
...	생략 부호는 특정 범위에서 생략된 항목을 나타낼 때 사용되기도 합니다.

관련 정보

LVM 및 HP-UX에 대한 추가 정보는 <http://docs.hp.com>의 **HP-UX Operating Environment** 모음에서 찾을 수 있습니다. 특히 다음 백서를 참조할 수 있습니다.

- LVM Limits
- LVM Migration from Legacy to Agile Naming Model
- LVM Online Disk Replacement (LVM OLR)
- LVM Version 2.0 Volume Groups in HP-UX 11i v3
- LVM Volume Group Dynamic LUN expansion (DLE) / vgmodify
- LVM Volume Group Quiesce/Resume
- SLVM Single-Node Online Reconfiguration (SLVM SNOR)
- When Good Disks Go Bad: Dealing with Disk Failures Under LVM

HP-UX 정보 찾기

다음 표에는 HP-UX에 대한 시스템 관리 정보를 찾을 수 있는 곳이 요약되어 있습니다. 이 표에는 특정 제품에 대한 정보는 들어 있지 않습니다.

표 1 HP-UX 정보 찾기

수행할 작업	참조 자료	위치
다음 내용 확인 <ul style="list-style-type: none"> • HP-UX 릴리즈에서 변경된 내용 • 운영 환경의 내용 • 특정 릴리즈의 펌웨어 요구 사항 및 지원되는 시스템 	HP-UX 버전에 해당하는 HP-UX 11i 릴리즈 노트	<ul style="list-style-type: none"> • HP Instant Information 미디어 • HP-UX 기술 문서 웹 사이트 http://docs.hp.com(영문) 및 http://docs.hp.com/ko(한글)
HP-UX 설치 또는 업데이트	<ul style="list-style-type: none"> • HP-UX를 설치 또는 업데이트하기 전에 • HP-UX 11i 설치 및 업데이트 설명서 	<ul style="list-style-type: none"> • 미디어 키트(운영 환경과 함께 제공됨) • HP Instant Information 미디어 • HP-UX 기술 문서 웹 사이트 http://docs.hp.com(영문) 및 http://docs.hp.com/ko(한글)
HP-UX 시스템 관리	HP-UX 11i v3 이후 릴리즈: <ul style="list-style-type: none"> • HP-UX 시스템 관리 설명서(다중 볼륨 모음) 기타 시스템 관리 정보 소스: <ul style="list-style-type: none"> • nPartition 관리 설명서 • Planning Superdome Configurations 백서 	<ul style="list-style-type: none"> • HP Instant Information 미디어 • HP-UX 기술 문서 웹 사이트 http://docs.hp.com(영문) 및 http://docs.hp.com/ko(한글) • “Planning Superdome Configurations”(http://docs.hp.com/hpux/onlinedocs/os/11i/superdome.pdf에서 제공)

HP-UX 11i 릴리즈 이름 및 운영 체제 버전 ID

HP-UX 11i는 중단 간 인터넷 컴퓨팅 환경의 요구 사항을 충족하는 가용성이 높고 보안이 뛰어나며 관리하기 쉬운 운영 체제입니다. HP-UX 11i는 기업에서 중요한 업무에 사용할 수 있는 기술적인 컴퓨팅 환경을 지원합니다. HP-UX 11i는 HP 9000 시스템과 HP Integrity 시스템에서 모두 사용할 수 있습니다.

각 HP-UX 11i 릴리즈에는 관련된 릴리즈 이름과 릴리즈 ID가 있습니다. `uname` 명령에 `-r` 옵션을 사용하면 릴리즈 ID가 반환됩니다. 표 2에서는 HP-UX 11i에서 사용 가능한 릴리즈를 보여 줍니다.

표 2 HP-UX 11i 릴리즈

OS 버전 ID	릴리즈 이름	지원되는 프로세서 아키텍처
B.11.11	HP-UX 11i v1	HP 9000
B.11.23	HP-UX 11i v2	Integrity
B.11.23.0409	HP-UX 11i v2 2004년 9월 업데이트	HP 9000 및 Integrity
B.11.31	HP-UX 11i v3	HP 9000 및 Integrity

HP-UX 11i의 각 버전에서 지원되는 시스템 및 프로세서 아키텍처에 대한 자세한 내용은 해당 HP-UX 버전의 HP-UX 11i 시스템 릴리즈 노트를 참조하십시오.

시스템 버전 확인

`uname`, `model` 및 `swlist` 명령을 사용하면 하드웨어 유형, 시스템 모델, 운영 체제 버전, 운영 환경 업데이트 상태를 비롯하여 시스템에 대한 정보를 확인할 수 있습니다. 자세한 내용은 `uname(1)`, `model(1)` 및 `swlist(1M)`를 참조하십시오.

OS 명령 규칙에 대해서는 “HP-UX 11i 릴리즈 이름 및 운영 체제 버전 ID” (15 페이지)를 참조하십시오.

표 3 OS 버전, 시스템 아키텍처 및 시스템 모델

주제	명령	출력 예
OS 버전	<code>\$ uname -r</code>	B.11.31 ¹
아키텍처	<code>\$ uname -m</code>	ia64 ² 9000/800 ²
시스템 모델	<code>\$ model</code> ³	ia64 hp server rx5670 9000/800/S16K-A
운영 환경	<code>\$ swlist HPUX*OE*</code>	# HPUX11i-OE-MC B.11.31 HP-UX Mission Critical Operating Environment ¹
OS 버전.업데이트	<code>\$ swlist HPUX*OE*</code>	# HPUX11i-TCOE B.11.23.0409 HP-UX Technical Computing OE Component ¹

1 HP-UX 11i OS 버전 ID의 형식은 B.11.23 또는 B.11.23.0409로, 여기서 B.11.23은 OS 버전이며 0409는 OE(운영 환경) 업데이트 연도-월입니다.

2 ia64 = Integrity. 기타 = HP 9000

3 `getconf MACHINE_MODEL` 명령은 동일한 출력을 제공합니다. 자세한 내용은 `getconf(1)`를 참조하십시오.

발행 정보

설명서 발행 날짜와 제품 번호로 설명서의 현재 버전을 확인할 수 있습니다. 새로운 버전의 설명서가 발행되면 발행 날짜가 변경됩니다. 설명서의 내용이 약간 변경되면 버전을 다시 찍을 때 발행 날짜가 변경되지 않습니다. 변경된 내용이 많은 경우 설명서의 번호가 변경됩니다. 다음 버전이 출판되기 전이라도 오류 수정이나 제품 변경에 따른 문서화를 위해 설명서가 업데이트될 수 있습니다. 새 버전의 설명서를 받으려면 해당 제품 지원 서비스에 가입해야 합니다. 자세한 사항은 HP 영업 담당자에게 문의하십시오. 다음 웹 사이트에서 이 설명서의 최신 버전을 찾을 수 있습니다.

<http://docs.hp.com/ko>

제품 번호:	지원되는 운영 체제	지원되는 버전	판 번호	발행일
5991-6493	HP-UX	HP-UX 11i v3	1	2007년 2월
5992-3400	HP-UX	HP-UX 11i v3	2	2008년 3월
5992-4592	HP-UX	HP-UX 11i v3	3	2008년 9월



참고: HP-UX 시스템 관리 설명서의 각 볼륨은 독립적으로 업데이트할 수 있습니다. 따라서 각 볼륨의 최신 버전이 시기별로 다를 수 있으며 서로 일치하지 않을 수도 있습니다. 각 볼륨의 최신 버전은 <http://docs.hp.com/ko>에서 볼 수 있습니다.

사용자 의견 접수

HP는 이 설명서에 대한 사용자 여러분의 의견을 기다리고 있습니다. HP는 사용자 요구를 충족하는 설명서를 만들기 위해 최선을 다할 것입니다. 발견한 오류, 개선을 위한 제안 사항 또는 격려의 말씀을 다음 주소로 보내 주십시오.

<http://docs.hp.com/ko/feedback.html>

의견을 보내실 때는 문서 제목, 제품 번호, 귀하의 의견, 발견한 오류 그리고 이 설명서를 개선하기 위해 도움이 될 만한 제안 사항 등을 함께 보내주십시오.

1 소개

이 장의 내용은 다음과 같습니다.

- “LVM 기능” (19 페이지)
- “LVM 아키텍처” (20 페이지)
- “물리 확장 영역과 논리 확장 영역 비교” (21 페이지)
- “LVM 볼륨 그룹 버전” (23 페이지)
- “LVM 장치 파일 사용” (24 페이지)
- “LVM 디스크 레이아웃” (28 페이지)
- “LVM 제한 사항” (30 페이지)

LVM 기능

LVM(Logical Volume Manager)은 파일 시스템이나 원시 데이터의 디스크 공간을 할당 및 관리할 수 있게 해주는 저장소 관리 시스템입니다. 이제까지는 디스크를 크기가 고정된 파티션으로 개별적으로 처리했습니다. 각 디스크 또는 파티션에는 파일 시스템, 스왑 공간, 부팅 영역 또는 원시 데이터를 저장했습니다. LVM을 사용하면 디스크나 크기가 고정된 파티션을 단일 용도로 지정하지 않아도 됩니다. 대신 디스크를 크기가 같은 확장 영역으로 구성된 데이터 저장소 풀(또는 볼륨)로 간주합니다. 확장 영역은 **논리 볼륨**이라는 가상 저장 장치에 할당되며, 논리 볼륨은 디스크처럼 취급할 수 있습니다.

LVM은 다음과 같은 기능을 제공합니다.

- 논리 볼륨의 크기는 변화하는 데이터 요구 사항을 충족하기 위해 동적으로 축소하거나 확장할 수 있습니다. 예를 들어, 논리 볼륨은 마운트되는 파일 시스템에서 필요한 만큼 작거나 클 수 있습니다. 파일 시스템은 해당 파일 시스템이나 논리 볼륨을 다시 구축하지 않고 확장할 수 있습니다. 파일 시스템을 축소하는 경우는 좀더 복잡하므로 파일 시스템을 다시 만들어야 할 수 있습니다.
- 여러 디스크에서 사용되지 않는 작은 공간들을 결합하여 사용 가능한 볼륨을 만들 수 있습니다.
- 논리 볼륨은 하나의 물리 디스크 크기를 초과할 수 있습니다. 단일 파일 시스템(및 개별 파일)이 여러 디스크에 걸쳐 있을 수 있기 때문에 이 기능을 **디스크 스페닝**이라고 합니다.
- LVM을 사용하여 동일한 데이터의 사본을 동시에 여섯 개까지 저장 및 업데이트할 수 있습니다. 이 기능은 논리 볼륨 **미러링**이라고 하며 선택적 제품인 HP MirrorDisk/UX가 필요합니다. 자세한 내용은 “미러링을 통해 데이터 가용성 늘리기” (38 페이지)를 참조하십시오.
- 미러 사본 중 하나가 실패하는 경우 별도의 디스크에 새 미러를 자동으로 만들도록 미러 데이터를 구성할 수 있습니다. 이 기능을 **스페어링**이라고 하며 선택적 제품인 HP MirrorDisk/UX가 필요합니다. 자세한 내용은 “디스크 스페어링을 통해 디스크 중복 늘리기” (42 페이지)를 참조하십시오.
- 논리적으로 연속된 데이터 블록(예: 동일한 파일의 청크)이 여러 디스크에 배포되도록 논리 볼륨을 만들 수 있습니다. 그러면 큰 파일을 순차적으로 읽고 쓸 때 입출

력 속도가 빨라집니다. 이 기능을 **스트라이핑**이라고 합니다. 스트라이핑과 미러링을 함께 사용할 수 있습니다. 자세한 내용은 “디스크 스트라이핑을 통한 성능 향상”(46 페이지)을 참조하십시오.

- 여러 링크를 통해 액세스한 장치를 구성하여 가용성을 높일 수 있습니다. 장치에 대한 기본 링크에 실패하면 LVM에서 대체 링크로 자동 전환할 수 있습니다. 이 기능을 **다중 경로 지정**이라고 합니다. 자세한 내용은 “다중 경로 지정을 통해 하드웨어 경로 중복 늘리기”(43 페이지)를 참조하십시오.

LVM 아키텍처

LVM 시스템은 LVM에 사용할 디스크를 초기화하는 것으로 시작합니다. LVM 디스크를 PV(**물리 볼륨**)라고 합니다. HP SMH(System Management Homepage) 또는 `pvccreate` 명령을 사용하여 디스크를 LVM 물리 볼륨으로 표시할 수 있습니다. 물리 볼륨은 일반적인 HP-UX 디스크 장치와 동일한 장치 특수 파일을 사용합니다.

LVM은 주소를 지정할 수 있는 PE(**물리 확장 영역**)라는 단위로 각 물리 볼륨을 나눕니다. 확장 영역은 디스크 시작 부분에 있는 LVM 메타데이터(인덱스 0)부터 시작하여 1씩 증분하여 순차적으로 할당됩니다. 물리 확장 영역 크기는 볼륨 그룹을 만들고 볼륨 그룹의 모든 디스크에 적용할 때 구성할 수 있습니다. 1MB에서 256MB 사이의 크기를 선택할 수 있습니다.

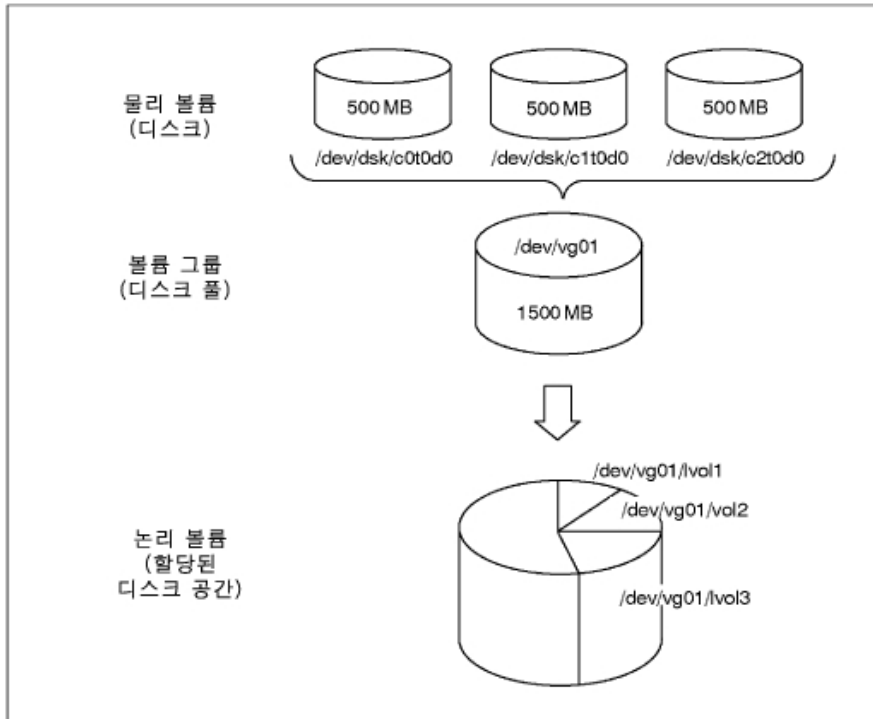
물리 볼륨은 VG(**볼륨 그룹**)로 구성됩니다. 볼륨 그룹은 하나 이상의 물리 볼륨으로 구성되며 시스템에는 두 개 이상의 볼륨 그룹이 있을 수 있습니다. 볼륨 그룹이 생성되면 디스크가 아닌 볼륨 그룹이 데이터 저장소를 나타내는 엔터티가 됩니다. 따라서 이전에는 한 시스템에서 다른 시스템으로 디스크를 옮겼지만, LVM에서는 한 시스템에서 다른 시스템으로 볼륨 그룹을 옮깁니다. 따라서 한 시스템에 볼륨 그룹을 여러 개 두는 것이 편리한 경우가 많습니다.

볼륨 그룹으로 표시되는 디스크 공간 풀은 다양한 크기의 LV(**논리 볼륨**)로 배분될 수 있습니다. 일단 논리 볼륨을 만들면 디스크 파티션처럼 취급할 수 있으며 장치 특수 파일을 통해 액세스할 수 있습니다. 논리 볼륨은 볼륨 그룹에 있는 여러 물리 볼륨에 걸쳐 있거나 한 물리 볼륨의 일부만 나타낼 수도 있습니다.

논리 볼륨의 기본 할당 단위를 LE(**논리 확장 영역**)라고 합니다. 논리 확장 영역은 물리 확장 영역에 매핑됩니다. 따라서 물리 확장 영역의 크기가 4MB면 논리 확장 영역의 크기도 4MB가 됩니다. 논리 볼륨의 크기는 구성된 논리 확장 영역의 수로 결정됩니다.

파일 시스템, 스왑, 덤프 또는 원시 데이터를 논리 볼륨에 지정합니다. 예를 들어, **그림 1-1**에서 보듯이 논리 볼륨 `/dev/vg01/lvol1`에는 파일 시스템이 들어가고, 논리 볼륨 `/dev/vg01/lvol2`에는 스왑 공간이 들어가고, 논리 볼륨 `/dev/vg01/lvol3`에는 원시 데이터가 들어갈 수 있습니다. HP SMH를 사용하여 지정된 크기의 논리 볼륨에 파일 시스템을 만든 다음 파일 시스템을 마운트합니다. 또는 LVM 명령을 사용하여 논리 볼륨을 만든 다음 확장하여 파일 시스템 또는 원시 데이터에 충분한 공간을 할당할 수 있습니다. 그런 다음 새 파일 시스템을 만들고 마운트하거나 논리 볼륨에 응용 프로그램을 설치합니다.

그림 1-1 논리 볼륨으로 파티션된 디스크 공간



물리 확장 영역과 논리 확장 영역 비교

LVM이 논리 볼륨에 디스크 공간을 할당할 때 물리 확장 영역에 대한 논리 확장 영역의 매핑이 자동으로 만들어집니다. 이 매핑은 논리 볼륨을 만들 때 선택한 정책에 따라 달라집니다. 논리 확장 영역은 0에서 시작하여 각 논리 볼륨에 순차적으로 할당됩니다. LVM에서는 데이터의 물리적인 위치에 관계없이 이 매핑을 사용하여 데이터에 액세스합니다. 이 매핑을 조사할 수 있는 명령이 있습니다. 자세한 내용은 `pvdiskplay(1M)` 및 `lvdisplay(1M)`를 참조하십시오.

미러, 스트라이프 또는 스트라이프-미러 논리 볼륨을 제외하고 각 논리 확장 영역은 한 물리 확장 영역에 매핑됩니다. 미러 논리 볼륨의 경우 각 논리 확장 영역은 미러 사본의 수에 따라 여러 물리 확장 영역에 매핑됩니다. 예를 들어, 미러 사본이 하나인 경우 각 논리 확장 영역은 두 물리 확장 영역에 매핑됩니다. 한 확장 영역은 원본에 대한 것이고 다른 하나는 미러 사본에 대한 것입니다. 미러링에 대한 자세한 내용은 “미러링을 통해 데이터 가용성 늘리기” (38 페이지)를 참조하십시오. 스트라이프 논리 볼륨에 대한 자세한 내용은 “디스크 스트라이핑을 통한 성능 향상” (46 페이지)을 참조하십시오. **Disk and File Management Tasks on HP-UX**라는 책도 참조하십시오.

그림 1-2에서는 볼륨 그룹의 물리 확장 영역과 논리 확장 영역 사이에서 사용 가능한 여러 종류의 매핑 예제를 보여 줍니다.

그림 1-2 물리 확장 영역 및 논리 확장 영역

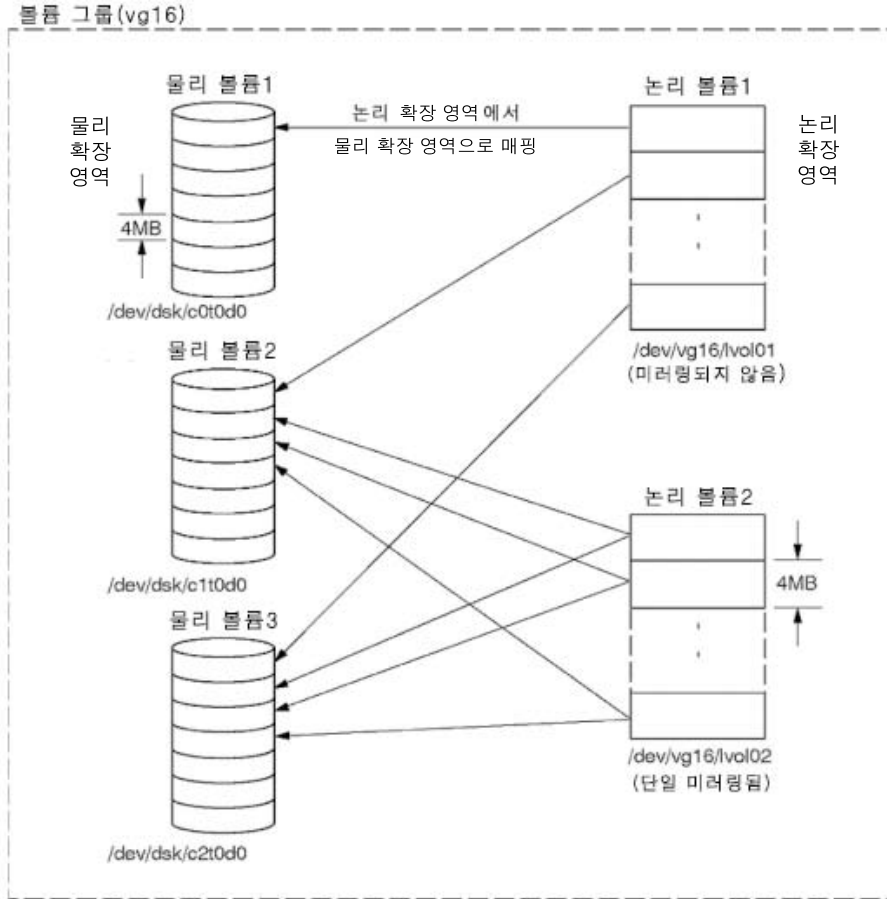


그림 1-2에 표시된 것과 같이 첫째 논리 볼륨의 내용은 볼륨 그룹의 세 물리 볼륨 모두에 들어 있습니다. 둘째 논리 볼륨은 미러링되었으므로 각 논리 확장 영역은 둘 이상의 물리 확장 영역에 매핑됩니다. 이 경우 데이터가 들어 있는 물리 확장 영역이 두 개이고 각 볼륨 그룹의 둘째 및 셋째 디스크에 있습니다.

기본적으로 LVM은 LVM 구성 파일인 `/etc/lvmtab` 및 `/etc/lvmtab_p`에 나타나는 순서에 따라 디스크에서 사용 가능한 물리 확장 영역을 선택하여 물리 확장 영역을 논리 볼륨에 지정합니다. 시스템 관리자는 이 기본 지정 절차를 건너뛰고 논리 볼륨에서 사용하는 디스크를 제어할 수 있습니다(“논리 볼륨을 특정 디스크로 확장” (66 페이지) 참조).

논리 볼륨이 루트, 부팅, 기본 스왑 또는 덤프에 사용될 경우 물리 확장 영역은 **연속적**이어야 합니다. 즉, 단일 물리 볼륨에 공백 없이 오름차순으로 물리 확장 영역을 할당해야 합니다. 루트, 부팅, 기본 스왑 또는 덤프에 사용되지 않는 논리 볼륨의 경우 논리 볼륨의 연속적인 논리 확장 영역에 대응되는 물리 확장 영역은 물리 볼륨에서 **연속적이지 않**거

나 전혀 다른 디스크에 상주할 수 있습니다. 결과적으로 한 논리 볼륨에서 만들어진 파일 시스템이 둘 이상의 디스크에 상주할 수 있습니다.

LVM 볼륨 그룹 버전

HP-UX 11i v3의 2008년 9월 릴리즈부터 LVM에서는 세 버전의 볼륨 그룹을 지원합니다. 특별히 언급된 경우가 아니라면 이 설명서의 모든 정보와 작업은 모든 볼륨 그룹 버전에 적용됩니다.

버전 1.0은 현재 및 이전 버전의 HP-UX 11i에서 모두 지원되는 버전입니다. 버전 1.0 볼륨 그룹을 관리하는 절차 및 명령 구문은 이전 릴리즈에서 변경되지 않습니다. 새 볼륨 그룹을 만들 때 `vgcreate`의 기본값은 버전 1.0으로 설정됩니다.

버전 2.0과 버전 2.1에서는 더 큰 볼륨 그룹, 논리 볼륨, 물리 볼륨 및 기타 매개 변수를 구성할 수 있습니다. 버전 2.1은 버전 2.0과 동일하지만, 사용할 수 있는 볼륨 그룹, 물리 볼륨 및 논리 볼륨의 수가 더 많습니다. 버전 2.x 볼륨 그룹은 을 제외하고는 버전 1.0 볼륨 그룹과 똑같이 관리됩니다.

- 버전 2.x 볼륨 그룹은 `vgcreate` 명령에 대한 옵션이 더 간단합니다. 버전 2.x 볼륨 그룹을 만들 때는 확장 영역 크기와 볼륨 그룹이 증가할 수 있는 최대 크기만 지정합니다. 이렇게 하면 공간을 관리할 때 LVM의 유연성이 향상됩니다. 작은 PV가 많은 볼륨 그룹과 큰 PV가 약간 있는 볼륨 그룹에 동일한 매개 변수를 사용할 수 있습니다. 볼륨 그룹 생성에 대한 자세한 내용은 “볼륨 그룹 만들기” (60 페이지)를 참조하십시오.
- HP-UX 11i v3의 2008년 3월 이전 버전을 포함하여 HP-UX 이전 릴리즈에서는 버전 2.0 볼륨 그룹이 인식되지 않으며, HP-UX 11i v3의 2008년 9월 이전 버전을 포함하여 HP-UX 이전 릴리즈에서는 버전 2.1 볼륨 그룹이 인식되지 않습니다.
- 버전 2.x 볼륨 그룹은 루트, 부팅, 스왑 또는 덤프 논리 볼륨을 지원하지 않습니다. `lvlnboot` 및 `lvrmboot` 명령을 버전 2.x 볼륨 그룹에서 실행하면 오류 메시지가 표시됩니다.
- 버전 2.x 볼륨 그룹은 부팅 가능한 물리 볼륨을 지원하지 않습니다. `pvccreate -B`를 사용하여 만든 물리 볼륨을 버전 2.x 볼륨 그룹에 추가할 수 없습니다.
- 버전 2.x 볼륨 그룹은 디스크 스페어링을 지원하지 않습니다. `vgextend` 또는 `pvchange` 명령에 `-z` 옵션을 사용하면 오류 메시지가 표시됩니다.
- `pvck` 및 `vgmodify` 명령은 버전 2.x 볼륨 그룹에서 지원되지 않습니다.
- 일부 HP-UX 제품에서는 버전 2.x 볼륨 그룹을 지원하지 않습니다. 자세한 내용은 현재 릴리즈에 대한 **HP-UX Logical Volume Manager and MirrorDisk/UX Release Notes**를 참조하십시오.

표 1-1에서는 버전 1.0과 버전 2.x 볼륨 그룹을 비교합니다.

표 1-1 LVM 볼륨 그룹 버전 등록 정보

	버전 1.0 볼륨 그룹	버전 2.0 볼륨 그룹	버전 2.1 볼륨 그룹
시스템 상의 최대 볼륨 그룹 수	256	512 ¹	2048 ¹
볼륨 그룹에 있는 최대 물리 볼륨 수	255	511	2048
볼륨 그룹에 있는 최대 논리 볼륨 수	255	511	2047
물리 볼륨의 최대 크기	2TB	16TB	16TB
볼륨 그룹의 최대 크기	510TB	2048TB	2048TB
논리 볼륨의 최대 크기	16TB	256TB	256TB
물리 확장 영역의 최대 크기	256MB	256MB	256MB
스트라이프의 최대 크기	32MB	256MB	256MB
최대 스트라이프 수	255	511	511
논리 볼륨당 최대 논리 확장 영역 수	65535	33554432	33554432
물리 볼륨당 최대 물리 확장 영역 수	65535	16777216	16777216
미러 사본 수 (MirrorDisk/UX 제품 필요)	0-2	0-5	0-5
LVM 구성 파일	/etc/lvmtab	/etc/lvmtab_p	/etc/lvmtab_p
장치 파일 주 번호	64	128	128

1 2048 볼륨 그룹 제한은 버전 2.0과 버전 2.1 볼륨 그룹에 모두 적용됩니다. 두 버전의 볼륨 그룹은 0-2047개 볼륨 그룹을 사용하여 만들 수 있지만, 버전 2.0 볼륨 그룹의 최대 작성 개수는 512입니다.

lvmdadm 명령을 사용하여 볼륨 그룹 제한을 표시할 수 있습니다. 자세한 내용은 **lvmdadm(1M)**을 참조하십시오.

LVM 장치 파일 사용

모든 LVM 구성 요소는 /dev 디렉토리에 있는 장치 특수 파일로 표시됩니다. 장치 특수 파일은 디스크 공간과의 상호 작용을 관리하는 에이전트로 생각할 수 있습니다. HP-PSMH와 HP-UX 명령을 둘 다 사용하여 LVM 장치 파일을 만듭니다. 이 절에서는 LVM에서 사용하는 장치 특수 파일 및 LVM 객체의 명명 규칙에 대해 설명합니다.

기존 장치 파일과 영구 장치 파일 비교

HP-UX 11i v3부터 디스크 장치를 /dev 디렉토리에서 두 개의 다른 장치 파일 유형인 **기존** 및 **영구** 유형으로 나타낼 수 있습니다.

기존 장치 파일은 HP-UX 11i v3 이전 릴리즈에서 대용량 저장 장치 파일의 유일한 유형이었습니다. 이 파일의 장치 파일 이름과 보조 번호에는 SCSI 버스, 대상, LUN 등의 하드웨어 경로 정보가 인코딩되어 있습니다. 예를 들어, 기존 장치 파일 /dev/dsk/c3t2d0은 카드 인스턴스 3, 대상 주소 2, LUN 주소 0인 디스크를 나타냅니다.

영구 장치 파일은 물리 하드웨어 경로를 디스크에 연결하는 대신 디스크의 고유한 WWID(WorldWide Identifier)에 매핑합니다. 따라서 디스크가 한 인터페이스에서 다른 인터페이스로 이동하거나, 한 스위치/허브에서 다른 스위치/허브로 이동하거나, 다른 대상 포트를 통해 호스트에 제공되어도 장치 파일이 변경되지 않습니다. 영구 장치 파일의 이름은 더 간단한 명명 규칙, /dev/disk/diskn을 따릅니다. 여기서 n은 디스크에 할당된 인스턴스 번호입니다. 장치 파일 이름이나 보조 번호에는 하드웨어 경로 정보가 포함되지 않습니다.

또한 디스크에 여러 하드웨어 경로가 있는 경우에는 단일 영구 장치 파일로 표시됩니다. 영구 장치 파일은 다중 경로가 지정된 디스크를 투명하게 처리하며 “**다중 경로 지정을 통해 하드웨어 경로 중복 늘리기**” (43 페이지)에 설명된 LVM의 다중 경로 지정 기능을 대체합니다. LVM에서 **pvlinks**라고 하는 하드웨어 경로가 디스크에 여러 개 있는 경우 영구 장치 특수 파일은 모든 링크의 단일 액세스 위치로 사용됩니다. I/O 요청은 선택한 로드 밸런싱 알고리즘에 따라 대용량 저장소 스택에서 사용 가능한 모든 링크 사이에 분배됩니다. 링크가 실패하면 대용량 저장소 스택에서 실패한 링크를 자동으로 비활성화하며, 나머지 링크 모두에서 I/O가 계속됩니다. 실패했거나 응답하지 않는 링크는 모니터링되므로 실패한 링크가 복구될 때 자동으로 그리고 투명하게 로드 밸런싱에 다시 통합됩니다. 또한 새 디스크와 링크는 자동으로 검색되어 로드 밸런싱에 추가됩니다. 링크 추가, 제거 또는 수정과 같은 디스크의 연결이 변경되더라도 하나 이상의 링크가 계속 활성 상태이면 영구 장치 파일을 사용하는 응용 프로그램에 영향을 주지 않습니다. 또한 새 디스크가 자동으로 검색됩니다.

LVM 디스크에 영구 또는 기존 장치 파일을 사용할 수 있습니다. 영구 장치 특수 파일이 더 다양한 로드 밸런싱 옵션을 지원하기 때문에 이 파일을 사용하는 것이 좋습니다.



참고: LVM의 대체 링크 기능을 사용하려면 기존 장치 파일을 사용하고 “**다중 경로 지정을 통해 하드웨어 경로 중복 늘리기**” (43 페이지)에 설명된 대로 기존 장치 파일을 통한 다중 경로 지정을 비활성화해야 합니다.

LVM의 명명 규칙

HP SMH 또는 HP-UX 명령에서 LVM 장치나 볼륨 그룹을 사용하는 경우 이름으로 참조해야 합니다. HP SMH와 HP-UX 명령으로 만들어지는 LVM 장치 파일은 기본적으로 표준 명명 규칙을 따릅니다. 하지만 볼륨 그룹과 논리 볼륨에 대해 사용자 정의된 이름을 선택할 수 있습니다.

물리 볼륨 이름

물리 볼륨은 해당 장치 파일 이름으로 식별됩니다. 예를 들면 다음과 같습니다.

표 1-2 물리 볼륨 명명 규칙

장치 파일 이름	장치 유형
/dev/disk/diskn	영구 블록 장치 파일
/dev/disk/diskn_p2	영구 블록 장치 파일, 파티션 2
/dev/rdisk/diskn	영구 특성 장치 파일
/dev/rdisk/diskn_p2	영구 특성 장치 파일, 파티션 2
/dev/dsk/cntndn	기존 블록 장치 파일
/dev/dsk/cntndns2	기존 블록 장치 파일, 파티션 2
/dev/rdsk/cntndn	기존 특성 장치 파일
/dev/rdsk/cntndns2	기존 특성 장치 파일, 파티션 2

각 디스크에는 블록 장치 파일과 특정 또는 원시 장치 파일(r 문자로 식별됨)이 있습니다. 사용하는 이름은 디스크에서 수행하는 작업에 따라 달라집니다.

HP Integrity 서버의 부팅 디스크에서는 반드시 _p2 접미사 또는 s2 접미사를 가지는 장치 파일을 사용하십시오. 이 접미사는 부팅 디스크상의 HP-UX 파티션을 의미합니다. HP 9000 서버에서는 파티션 번호가 없는 장치 파일을 사용하십시오.

다음 작업에만 물리 볼륨의 원시 장치 파일을 사용하십시오.

- pvcreate 명령을 사용하여 LVM의 물리 볼륨을 준비하는 경우. 이 경우 디스크에 장치 파일을 사용합니다(예: /dev/rdisk/disk14). 파티션 접미사가 없는 것은 전체 디스크를 참조하는 경우를 나타냅니다.
- pvremove 명령을 사용하여 물리 볼륨에서 LVM 정보를 제거하는 경우
- vgcfgrestore 명령을 사용하여 볼륨 그룹 구성을 복원하는 경우
- pvck 명령을 사용하여 물리 볼륨에서 일관성 검사를 수행하는 경우
- vgchgid 명령을 사용하여 물리 볼륨에서 볼륨 그룹 식별자를 수정하는 경우
- vgmodify 명령을 사용하여 물리 볼륨의 디스크 유형을 변경하는 경우

다른 작업에는 모두 블록 장치 파일을 사용하십시오. 예를 들어, vgextend 명령을 사용하여 볼륨 그룹에 물리 볼륨을 추가하는 경우 /dev/disk/disk14와 같은 디스크의 블록 장치 파일을 사용합니다.

새 디스크가 검색되면 자동으로 디스크 장치 파일이 모두 만들어집니다. 자세한 내용은 insf(1M)를 참조하십시오.

볼륨 그룹 이름

각 볼륨 그룹의 이름은 고유해야 하며 255자 이하여야 합니다. 예를 들어 일반적인 볼륨 그룹 이름은 vg01, vgroot 또는 vg_sales가 될 수 있습니다. 이름이 vg로 시작할 필

요는 없지만 이 접두사를 사용하는 것이 좋습니다. 기본적으로 HP SMH에서는 /dev/vg_{nn} 형식의 이름을 사용합니다. 숫자 _{nn}은 00에서 시작하여 볼륨 그룹이 만들어지는 순서대로 증가합니다. 기본적으로 루트 볼륨 그룹은 vg00입니다.

논리 볼륨 이름

논리 볼륨은 해당 장치 파일 이름으로 식별됩니다. 이 이름은 사용자가 지정하거나 논리 볼륨을 만들 때 lvcreate 명령을 사용하여 기본적으로 지정할 수 있습니다.

사용자가 지정할 경우 255자 이하의 이름을 원하는 대로 선택할 수 있습니다.

기본적으로 지정할 경우 이러한 이름은 /dev/vg_{nn}/lv_{volN}(블록 장치 파일 형태) 및 /dev/vg_{nn}/rl_{volN}(특성 장치 파일 형태)의 형태를 지닙니다. 숫자 _N은 1에서 시작하며 각 볼륨 그룹에서 논리 볼륨이 만들어지는 순서대로 증가합니다.

LVM이 논리 볼륨을 만들 때 블록 및 특성 장치 파일을 모두 만듭니다. 그런 다음 LVM은 논리 볼륨에 대한 장치 파일을 적절한 볼륨 그룹 디렉토리에 저장합니다. 예를 들어 볼륨 그룹 vg01에 만들어진 첫 번째 논리 볼륨의 기본 블록 이름은 다음과 같은 전체 경로 이름을 갖습니다.

```
/dev/vg01/lvol1
```

판매 데이터베이스가 포함된 논리 볼륨을 만든 경우 다음과 같이 명시적으로 이름을 지정할 수 있습니다.

```
/dev/vg01/sales_db_lv
```

위 예제의 논리 볼륨을 만든 후 해당 볼륨은 두 장치 파일을 갖게 됩니다. 블록 장치 파일에 대한 /dev/vg01/sales_db_lv와 특성 또는 원시 장치 파일에 대한 /dev/vg01/rsales_db_lv 장치 파일을 갖습니다.

물리 볼륨 그룹 이름

물리 볼륨 그룹은 미러링에 유용합니다. 자세한 내용은 “I/O 채널 분리를 통한 성능 향상”(48 페이지)을 참조하십시오. 이 경우 유일한 이름 지정 제한은 볼륨 그룹에서 각 물리 볼륨 그룹의 이름이 고유해야 한다는 것입니다. 예를 들어, 볼륨 그룹 /dev/vg02에 pv_{g1} 및 pv_{g2}라는 두 물리 볼륨 그룹이 있을 수 있습니다.

장치 번호 형식

LVM과 관련된 장치 파일은 /dev 디렉토리에 상주합니다. 각 볼륨 그룹의 경우 /dev 아래의 디렉토리 이름은 볼륨 그룹에 따라 지정됩니다. 해당 디렉토리에는 하나의 "group" 장치 파일과 각 논리 그룹에 대한 개별 블록 및 특성 장치 파일이 있습니다.

다음은 샘플 목록입니다.

```
# ls -l /dev/vg01
total 0
crw-r--r--  1 root      root          64 0x010000 Mar 28  2004 group
brw-r-----  1 root      root          64 0x010001 Jul 29 16:53 lvol1
brw-r-----  1 root      root          64 0x010002 Jul 29 16:53 lvol2
crw-r-----  1 root      root          64 0x010001 Mar 28  2004 rlvol1
crw-r-----  1 root      root          64 0x010002 Mar 28  2004 rlvol2
```

기본적으로 볼륨 그룹 번호는 0(vg00)으로 시작하지만 논리 볼륨은 1(lvol1)로 시작합니다. 그 이유는 논리 볼륨 번호가 보조 번호와 일치하고 볼륨 그룹의 그룹 파일에 보조 번호 0이 할당되기 때문입니다.

물리 볼륨은 해당 디스크와 관련된 장치 파일을 사용합니다. LVM은 물리 볼륨에 대한 장치 파일을 만들지 않습니다.

버전 1.0 장치 번호 형식

표 1-3에는 버전 1.0 볼륨 그룹에 대한 장치 파일 번호의 형식이 나열되어 있습니다.

표 1-3 버전 1.0 장치 번호 형식

주 번호	볼륨 그룹 번호	예약됨	논리 볼륨 번호
64	0-0xff	0	0-0xff 0=그룹 파일

버전 1.0 볼륨 그룹의 경우 LVM 장치 파일의 주 번호는 64입니다. 볼륨 그룹 번호는 보조 번호의 상위 8비트로 인코딩되고 논리 볼륨 번호는 하위 8비트로 인코딩됩니다. 논리 볼륨 번호 0은 그룹 파일용으로 예약되어 있습니다.

버전 2.x 장치 번호 형식

표 1-4에는 버전 2.x 볼륨 그룹에 대한 장치 파일 번호의 형식이 나열되어 있습니다.

표 1-4 버전 2.x 장치 번호 형식

주 번호	볼륨 그룹 번호	논리 볼륨 번호
128	0-0x7ff	0-0x7ff 0=그룹 파일

버전 2.x 볼륨 그룹의 경우 LVM 장치 파일의 주 번호는 128입니다. 볼륨 그룹 번호는 보조 번호의 상위 12비트로 인코딩되고 논리 볼륨 번호는 하위 12비트로 인코딩됩니다. 논리 볼륨 번호 0은 그룹 파일용으로 예약되어 있습니다.



참고: 볼륨 그룹 번호와 논리 볼륨 번호 필드의 가장 중요한 비트는 예약되어 있으며 0이어야 합니다.

장치 번호 형식은 변경될 수 있습니다.

LVM 디스크 레이아웃



참고: 이 정보는 버전 1.0 볼륨 그룹에 속하는 디스크에만 적용됩니다.

LVM 디스크 레이아웃은 두 종류로 부팅 디스크용과 다른 모든 LVM 디스크용이며, 이들은 데이터 구조가 서로 다릅니다. 부팅 불가능한 디스크에는 PVRA(Physical Volume Reserved Area)와 VGRA(Volume Group Reserved Area)라는 두 개의 예약 영역이 있

습니다. 부팅 가능 디스크에는 PVRA, VGRA 및 BDRA(Boot Data Reserved Area)와 부팅 LIF에 예약된 추가 섹터가 있습니다.

Boot Data Reserved Area

BDRA는 루트, 기본 스왑 및 덤프 논리 볼륨을 구성하고 루트 파일 시스템을 마운트할 때 필요한 정보를 갖고 있습니다.

BDRA에서 LVM 디스크 데이터 구조 정보는 `lvlnboot` 및 `lvrmboot` 명령을 사용하여 유지 관리됩니다. 다음은 샘플 출력입니다.

```
# lvlnboot -v
볼륨 그룹 /dev/vg00에 대한 부팅 정의:
루트 볼륨 그룹에 속하는 물리 볼륨::
    /dev/dsk/c3t0d0 -- 부팅 디스크
    /dev/dsk/c4t0d0 -- 부팅 디스크
    /dev/dsk/c5t0d0
    /dev/dsk/c12t0d0 -- 부팅 디스크
루트: lv011      on: /dev/dsk/c3t0d0
                   /dev/dsk/c4t0d0
스왑: lv012      on: /dev/dsk/c3t0d0
                   /dev/dsk/c4t0d0
덤프: lv012      on: /dev/dsk/c3t0d0
mkboot 및 pvcreate -B로 초기화되어 "부팅 디스크"로 지정된 물리 볼륨은 부팅 가능
합니다. lv011 및 lv012에 대한 여러 줄은 루트 및 스왑 논리 볼륨이 미러링 중임을
나타냅니다.
```

Logical Interface Format Area

LVM 부팅 디스크에는 LABEL 파일이 저장되는 LIF(Logical Interface Format) 영역이 있습니다. HP 9000 서버의 LIF 영역에는 초기 시스템 로더(ISL), 커널 부팅 로더(HPUX) 및 자동 부팅 파일(AUTO)과 같은 부팅 유틸리티 및 오프라인 진단 프로그램이 들어 있습니다.

LABEL 파일은 `lvlnboot` 및 `lvrmboot`를 통해 만들고 유지 관리합니다. 이 파일에는 부팅 파일 시스템(/stand)을 포함하여 부팅 관련 논리 볼륨의 시작점 및 크기 정보가 들어 있습니다. 유틸리티는 실제로 LVM을 사용하지 않고 LABEL 파일을 사용하여 루트, 기본 스왑 및 덤프 논리 볼륨에 액세스할 수 있습니다.

Physical Volume Reserved Area

PVRA(Physical Volume Reserved Area)에는 고유 식별자 및 물리 확장 영역 정보와 같은 물리 볼륨을 설명하는 정보 및 디스크에 있는 다른 LVM 구조에 대한 포인터가 들어 있습니다.

Volume Group Reserved Area

VGRA(Volume Group Reserved Area)는 디스크가 속한 볼륨 그룹에 대해 설명합니다. 이 정보는 모든 물리 볼륨에 복제되고 구성이 변경될 때마다 업데이트됩니다. 이 정보는 다른 데이터와 함께 다음 정보를 포함합니다.

- 물리 볼륨 상태 및 크기, 논리 볼륨에 대한 물리 확장 영역의 맵을 비롯한 볼륨 그룹의 물리 볼륨 목록
- 각 논리 볼륨의 상태 및 기능, 일정 및 할당 정책, 미러 사본 수를 비롯한 볼륨 그룹의 논리 볼륨 목록
- VGID(Volume Group Identifier)와 다음 세 개의 구성 가능한 매개 변수를 포함한 볼륨 그룹 헤더
 - 볼륨 그룹에서 허용되는 물리 볼륨 수
 - 볼륨 그룹에서 허용되는 논리 볼륨의 최대 개수
 - 물리 볼륨당 허용되는 물리 확장 영역의 최대 개수

각 물리 확장 영역은 VGRA에 기록되므로 확장 영역 크기는 VGRA 크기와 직접 관계가 있습니다. 대부분의 경우 기본 확장 영역 크기는 충분합니다. 그러나 문제가 발생할 경우 VGRA가 고정 크기이고 대용량 물리 볼륨이 허용된 총 물리 확장 영역 수를 초과할 수 있다는 점을 고려해야 합니다. 따라서 대용량 LVM 디스크의 기본 확장 영역보다 큰 크기를 사용해야 할 수도 있습니다. 반대로 볼륨 그룹의 모든 LVM 디스크가 작으면 기본 확장 영역 수로 인해 VGRA가 너무 커져서 디스크와 메모리 공간을 낭비할 수 있습니다. 물리 확장 영역의 기본 확장 영역 크기나 개수보다 적게 사용하는 것이 더 좋습니다. 대용량 물리 볼륨은 확장 영역 크기가 작거나 디스크당 물리 확장 영역 수가 적게 설정된 볼륨 그룹에서 사용하지 못할 수 있습니다.

사용자 데이터 영역

사용자 데이터 영역은 파일 시스템, 가상 메모리 시스템(스왑) 또는 사용자 응용 프로그램을 비롯하여 모든 사용자 데이터를 저장하는 데 사용되는 LVM 디스크의 영역입니다.

LVM 제한 사항

LVM은 복잡한 하위 시스템이므로 배우는 데 시간이 걸리고 유지 관리가 필요하며 아주 드문 경우지만 문제가 발생할 수 있습니다.

논리 볼륨을 기본 디스크 관리 방법으로 사용하는 것이 좋습니다. 파일 및 응용 프로그램 서버에서는 LVM을 사용합니다. 운영 체제 저장과 스왑용으로만 사용되는 디스크가 하나뿐인 서버에서는 "전체 디스크" 관리 방법을 사용하는 것이 더 쉽고 간단합니다. 이러한 시스템에는 LVM이 필요하지 않습니다.

기본적으로 LVM 구성은 `/etc/lvmconf`에서 변경할 때마다 자동으로 백업됩니다. 전체 디스크 방법에서는 사용할 수 없는 미러링은 데이터 손실을 방지합니다.

LVM에 대한 추가 제한은 다음과 같습니다.

- LVM 디스크 및 비LVM 디스크는 시스템에 동시에 존재할 수 있지만 지정한 디스크나 파티션은 LVM 또는 비LVM 방법 중 하나를 선택하여 전체를 관리해야 합니다. 즉, 하나의 디스크나 파티션에 이러한 기술을 함께 사용할 수 없습니다.

- HP Integrity 서버에서는 LVM이 루트 디스크 및 해당 미러의 파티션 지정만 지원하며 모든 디스크에서 하나의 HP-UX 파티션만 지원합니다.
- 플로피 디스크, 광학 디스크 및 CD-ROM에서는 논리 볼륨이 지원되지 않습니다.
- 루트 디스크에는 LVM 또는 VxVM(VERITAS™ Volume Manager) 디스크를 사용해야 합니다.
- LVM을 사용하려면 먼저 디스크를 물리 볼륨으로 초기화해야 합니다.
- 물리 볼륨을 저장소에 대해 할당하려면 볼륨 그룹에 지정되어 있어야 합니다.
- 한 물리 그룹은 한 볼륨 그룹에만 속할 수 있습니다.
- 볼륨 그룹의 확장 영역 크기는 볼륨 그룹을 만들 때 고정됩니다. 따라서 변경하려면 볼륨 그룹을 다시 만들어야 합니다.

2 LVM 구성

기본적으로 LVM 명령은 시스템에 이미 설치되어 있습니다. 이 장에서는 논리 볼륨을 설정할 때 고려해야 할 문제에 대해 설명합니다. 다음 내용에 대해 설명합니다.

- “LVM 구성 계획” (33 페이지)
- “여러 유형의 논리 볼륨 설정” (33 페이지)
- “가용성 계획” (37 페이지)
- “복구 계획” (49 페이지)
- “성능 계획” (44 페이지)

LVM 구성 계획

논리 볼륨을 사용하려면 몇 가지 계획이 필요합니다. 이 장에서는 계획할 때 고려할 몇 가지 문제에 대해 설명합니다. 이러한 문제는 시스템에 논리 볼륨을 설정하거나 수정하기 전에 고려해야 합니다.

- 논리 볼륨을 사용하는 목적은 무엇입니까? 원시 데이터나 파일 시스템에 사용합니까? 스왑 영역이나 덤프 영역으로 사용합니까? 자세한 내용은 “여러 유형의 논리 볼륨 설정” (33 페이지)을 참조하십시오.
- 논리 볼륨을 얼마나 크게 만들어야 합니까?
- 고가용성 데이터여야 합니까? 그렇다면 “미러링을 통해 데이터 가용성 늘리기” (38 페이지)에 설명된 대로 여러 디스크에 걸쳐 논리 볼륨을 미러링합니다. 또는 “디스크 스페어링을 통해 디스크 중복 늘리기” (42 페이지)에 설명된 대로 미러 오류를 처리할 수 있도록 스페어 디스크를 설정합니다.
- I/O 성능이 매우 중요한 문제입니까? 그렇다면 “디스크 스트라이핑을 통한 성능 향상” (46 페이지)에 설명된 대로 여러 디스크에 걸쳐 논리 볼륨을 스트라이핑하고 “I/O 채널 분리를 통한 성능 향상” (48 페이지)에 설명된 대로 I/O 채널을 분리합니다. 성능에 관한 추가 권장 사항은 “성능 계획” (44 페이지)을 참조하십시오.
- 고가용성과 I/O 성능 사이에서 균형을 찾아야 합니까? 그렇다면 “미러링을 통해 데이터 가용성 늘리기” (38 페이지) 및 “디스크 스트라이핑을 통한 성능 향상” (46 페이지)에 설명된 대로 논리 볼륨을 스트라이핑하고 미러링합니다.
- 디스크 장애를 빨리 복구하는 것이 중요합니까? 그렇다면 “복구 계획” (49 페이지)을 참조하십시오.

여러 유형의 논리 볼륨 설정

이 절에서는 특수 논리 볼륨의 설정에 대해 설명합니다.

원시 데이터 저장소용 논리 볼륨 설정

특히 원시 데이터 저장소용 논리 볼륨을 만들어 원시 I/O 성능을 최적화할 수 있습니다. 데이터베이스와 같은 원시 데이터 논리 볼륨을 만들려면 논리 볼륨 크기와 디스크에 논리 볼륨을 분배할 방법을 고려해야 합니다.

일반적으로 논리 볼륨의 크기는 MB로 지정합니다. 그러나 논리 볼륨의 크기는 볼륨 그룹에 사용된 확장 영역 크기의 배수여야 합니다. 예를 들어 데이터베이스 파티션에 2002MB가 필요하고 논리 확장 영역 크기가 4MB이면 LVM에서는 2004MB의 논리 볼륨을 만들거나 논리 확장 영역 501개를 만듭니다.

데이터베이스 파티션 설치와 같이 원시 데이터 저장소 논리 볼륨을 많이 사용할 경우 디스크에 논리 볼륨이 분배되는 방식을 고려해야 합니다.

기본적으로 LVM은 논리 볼륨의 디스크 공간을 한 물리 볼륨에서 지정하고 이 물리 볼륨의 공간을 모두 사용한 다음 같은 방식으로 그 다음 물리 볼륨에 공간을 지정합니다. LVM은 `/etc/lvmtab` 및 `/etc/lvmtab_p`에 나타나는 순서대로 물리 볼륨을 사용합니다. 즉, 논리 볼륨의 데이터가 볼륨 그룹 내의 모든 물리 볼륨에 골고루 분배되지 않을 수도 있습니다.

따라서 논리 볼륨에 I/O 액세스가 발생하면 볼륨 그룹 내에 있는 하나 이상의 디스크가 과도하게 사용되는 반면 다른 디스크는 적게 사용되거나 전혀 사용되지 않을 수 있습니다. 이러한 배열 방식으로는 최적의 I/O 성능을 얻을 수 없습니다.

다른 더 좋은 방법에는 논리 볼륨을 특정 디스크에 인터리브 방식으로 설치하는 방법이 있습니다. 이렇게 하면 I/O 액세스가 골고루 분배되고 성능이 최적화됩니다(“논리 볼륨 확장” (65 페이지) 참조).

논리 볼륨의 내용이 원시 데이터에 사용되고 있는지 식별할 수 있는 HP-UX 명령이 없으므로 원시 데이터에 대해 만드는 논리 볼륨에 쉽게 인식할 수 있는 이름을 지정하는 것이 좋습니다. 이렇게 하면 논리 볼륨의 내용을 인식할 수 있습니다.

파일 시스템용 논리 볼륨 설정

파일 시스템은 디스크 파티션 또는 파티션되지 않은 디스크에 상주하는 것처럼 논리 볼륨에 상주합니다. 논리 볼륨에서는 두 가지 유형의 파일 시스템, 즉 HFS(Hierarchical File System)와 JFS(Journaled File System)(VxFS)를 사용할 수 있습니다.

파일 시스템 논리 볼륨의 초기 크기 선택

파일 시스템에 필요한 공간을 결정할 때에는 그림 2-1에 표시된 세 가지 주요 구성 요소를 고려해야 합니다.

그림 2-1 파일 시스템 공간 구성 요소



파일 시스템을 포함할 논리 볼륨의 크기를 추정하려면 다음 단계를 수행하십시오.

1. 앞으로 사용자가 데이터를 사용하는 데 필요하게 될 디스크 공간을 추정합니다. 일반적으로 추가 성장할 것으로 기대되는 변화량을 감안합니다. 현재 사용되고 있는 디스크 공간을 보려면 `du` 명령을 사용합니다.
2. 위의 “minfree” 영역에 대한 크기에 10%를 추가합니다. 이 영역은 성능 유지 관리를 위해 예약되어 있습니다.
3. 파일 시스템 오버헤드로 5%를 더 추가합니다. 여기에는 파일 시스템을 유지 관리하는 데 필요한 모든 데이터 구조가 포함됩니다.
4. 이 논리 볼륨에 사용된 논리 확장 영역 크기의 다음 정수 배수로 올려 논리 확장 영역에서의 크기를 구합니다. 이 단계는 논리 볼륨을 만들 때 자동으로 수행됩니다.

예를 들어 사용자 그룹이 파일 시스템 데이터에 사용할 공간으로 60MB 필요한 경우 다음은 앞으로 성장할 것으로 기대되는 양을 감안한 것입니다. minfree 공간으로 6MB를 추가합니다. 파일 시스템 오버헤드로 3MB를 더 추가하면 파일 시스템에 필요한 전체 크기는 69MB가 됩니다. 이것은 파일 시스템이 포함되는 논리 볼륨의 크기입니다. 확장 영역 크기가 4MB인 볼륨 그룹에 논리 볼륨을 만들면 69는 4MB로 나눌 수 있는 72로 올려 집니다.

측정값이 정밀하지 않더라도 파일 시스템 크기를 계획하는 데는 충분합니다. 크기를 늘리기 전에 어느 정도 유용하게 사용할 수 있을 만큼 충분히 크게 파일 시스템을 만드십시오.



팁: 일반적으로 파일 시스템의 크기를 늘리는 것이 줄이는 것보다 쉬우므로 파일 시스템을 만들 크기를 신중하게 추정하는 것이 좋습니다.

루트 파일 시스템의 경우는 예외입니다. 연속 논리 볼륨인 루트 파일 시스템은 확장하기가 어렵습니다.

파일 시스템 논리 볼륨 크기 조정

사용자가 파일 시스템에 원래 할당된 공간을 초과하여 사용한 경우 우선 `lvextend` 명령을 사용하여 파일 시스템이 상주하는 논리 볼륨을 확장한 다음 `extendfs` 명령을 사용하여 논리 볼륨에 포함된 파일 시스템을 늘려서 파일 시스템의 크기를 늘릴 수 있습니다.

파일 시스템의 크기를 줄이는 일은 어려울 수 있습니다. 파일 시스템의 유형에 따라 크기를 줄이지 못할 수도 있습니다. 그러나 더 작은 **새로운** 파일 시스템을 만들어 교체할 수는 있습니다.

파일 시스템 논리 볼륨의 크기 조정에 대한 자세한 내용은 “파일 시스템 논리 볼륨 관리”(105 페이지)를 참조하십시오.

파일 시스템 논리 볼륨 지침

파일 시스템 논리 볼륨을 구성할 경우 다음 지침을 따르십시오.

- 여러 LVM 디스크로 확장되는 파일 시스템을 만드는 경우 최상의 시스템 성능을 얻으려면 파일 시스템이 있는 논리 볼륨이 동일한 디스크 유형으로 확장되어야 합니다.
- LVM은 기본적으로 성능을 고려하지 않고 사용 가능한 디스크에 논리 볼륨을 만듭니다. 파일 시스템을 특성이 다른 두 디스크에 확장할 수 있지만 이 경우 파일 시스템 성능이 나빠질 수 있습니다.

다음 두 단계를 수행하여 논리 볼륨의 물리 확장 영역이 들어갈 물리 볼륨을 제어할 수 있습니다.

1. `lvcreate` 명령 또는 HP SMH를 사용하여 크기를 지정하지 않고 논리 볼륨을 만듭니다. 크기를 지정하지 않으면 기본적으로 논리 볼륨에 물리 확장 영역이 할당되지 않습니다.
 2. 이제 `lvextend` 명령을 사용하여 파일 시스템이 들어갈 특정 물리 볼륨으로 논리 볼륨을 확장합니다. 즉, 공간을 할당합니다.
- 루트 또는 부팅 논리 볼륨은 프로세서에 따라 2GB 또는 4GB로 제한됩니다.

스왑용 논리 볼륨 설정



참고: 버전 2.x 볼륨 그룹은 스왑 논리 볼륨을 지원하지 않습니다.

이 절에서는 논리 볼륨을 스왑 장치로 사용하는 경우에 고려해야 할 사항에 대해 설명합니다. 시스템에 필요한 스왑 공간의 크기와 유형 결정 등 시스템 스왑 공간 관리에 대한 자세한 내용은 **HP-UX 시스템 관리 설명서: 구성 관리**를 참조하십시오.

스왑으로 구성된 논리 볼륨은 **장치 스왑 공간**으로 취급됩니다. 장치 스왑 공간은 논리 볼륨 또는 파티션에 위치하며 일반적으로 스와핑을 위해 명시적으로 예약됩니다. 이 공간은 덤프 영역으로 구성될 수도 있습니다(**“덤프 논리 볼륨 지침”** (37 페이지) 참조).

스왑 논리 볼륨 지침

스왑 논리 볼륨을 구성할 경우 다음 지침을 따르십시오.

- 성능을 더 좋게 하기 위해 장치 스왑 영역을 인터리브합니다.
공간의 크기가 같을 경우 한 스왑 영역에서보다 서로 다른 디스크에 있는 두 스왑 영역에서 성능이 좋습니다. 이 구성에서는 **인터리브** 스와핑이 가능합니다. 즉, 동시에 스왑 영역에 쓰여지므로 성능이 향상됩니다.
LVM을 사용하는 경우 `lvextend`를 사용하여 서로 다른 디스크에 있는 논리 볼륨에 보조 스왑 영역을 설치해야 합니다.
디스크가 하나만 있고 스왑 공간을 증가시켜야 하는 경우 기본 스왑 영역을 보다 큰 영역으로 이동합니다.

- 비슷한 크기의 장치 스왑 영역에서 가장 잘 작동합니다.
최상의 성능을 얻으려면 장치 스왑 영역의 크기가 비슷해야 합니다. 그렇지 않으면 더 작은 장치 스왑 영역의 모든 공간이 사용되고 더 큰 스왑 영역만 사용할 수 있는 경우 인터리빙할 수 없게 됩니다.
- 기본적으로 기본 스왑은 루트 파일 시스템과 같은 디스크에 있습니다. 기본적으로 시스템 커널 구성 파일 `/stand/system`에 기본 스왑에 대한 구성 정보가 들어 있습니다.
- 논리 볼륨을 보조 스왑으로 사용하고 있는 경우에는 보조 스왑을 루트 디스크가 아닌 디스크에 상주하도록 할당하여 성능을 향상시킵니다.

덤프용 논리 볼륨 설정



참고: 버전 2.x 볼륨 그룹은 덤프 논리 볼륨을 지원하지 않습니다.

이 절에서는 논리 볼륨을 덤프 장치로 사용하는 경우에 고려해야 할 사항에 대해 설명합니다. 덤프 영역은 시스템이 손상된 후 코어 메모리의 이미지를 작성하는 데 사용되는 디스크 공간입니다. 코어 덤프 분석은 문제를 해결하거나 시스템을 작동 상태로 복원하는 데 유용할 수 있습니다.

기본적으로 기본 스왑 장치는 덤프 영역이 특별히 지정되지 않은 경우 덤프 영역의 역할도 수행합니다. 기본 스왑을 덤프 영역으로 유지할 필요는 없지만 이렇게 하면 디스크 공간이 유지됩니다. 시스템에 다른 덤프 장치 또는 여러 덤프 장치를 구성할 수 있습니다. 이렇게 하려면 논리 볼륨을 덤프 장치로 만들어야 합니다. 이 장치는 스왑에도 사용할 수 있습니다.

덤프 장치의 추가, 제거 또는 수정 및 덤프 알고리즘 구성에 대한 자세한 내용은 **HP-UX 시스템 관리 설명서: 구성 관리**를 참조하십시오.

덤프 논리 볼륨 지침

덤프 논리 볼륨을 구성할 경우 다음 지침을 따르십시오.

- 디스크 파티션이 아닌 덤프 영역에 논리 볼륨을 사용하는 것이 좋습니다.
- 덤프 논리 볼륨은 루트 볼륨 그룹, 즉 루트 논리 볼륨이 들어 있는 볼륨 그룹에만 존재할 수 있습니다.
- 스왑 영역이 루트 볼륨 그룹에 있으면 모든 보조 스왑 논리 볼륨을 덤프 영역으로 사용할 수 있습니다.

가용성 계획

이 절에서는 가용성 및 데이터 중복을 향상시킬 수 있는 LVM 기능에 대해 설명합니다. 다음 내용에 대해 설명합니다.

- “미러링을 통해 데이터 가용성 늘리기” (38 페이지)
- “디스크 스페어링을 통해 디스크 중복 늘리기” (42 페이지)
- “다중 경로 지정을 통해 하드웨어 경로 중복 늘리기” (43 페이지)

미러링을 통해 데이터 가용성 늘리기



참고: 미러링을 사용하려면 선택적 제품인 HP MirrorDisk/UX가 필요합니다.

미러링은 동일한 데이터의 사본을 논리 볼륨에 저장하는 기능을 의미합니다. 이때 데이터 사본은 서로 다른 디스크에 저장하는 것이 좋습니다. 이러한 중복에는 몇 가지 장점이 있습니다.

- 루트 파일 시스템과 스왑을 미러링하면 중요한 데이터를 두 개 이상의 LVM 디스크에서 사용할 수 있으므로 운영 체제에서 루트 디스크 장애를 극복할 수 있습니다.
- 특정 응용 프로그램에서 사용하는 논리 볼륨을 미러링하면 디스크 장애가 발생해도 응용 프로그램은 계속 실행됩니다.
- I/O 채널에 장애가 발생해도 LVM은 중복된 소스에서 데이터를 복구할 수 있습니다.
- 미러링을 사용하면 하드웨어가 가장 편리한 LVM 디스크에서 데이터를 읽어 I/O를 최적화하므로 읽기 작업이 많은 응용 프로그램의 속도를 높일 수 있습니다.
- 다른 사본이 계속 실행되는 동안 데이터 사본 하나를 백업할 수 있습니다.

미러링을 사용하면 하나의 논리 확장 영역이 둘 이상의 물리 확장 영역 집합에 매핑됩니다. 논리 확장 영역의 수는 그대로 유지되지만 사용된 물리 확장 영역(따라서 사용된 디스크 공간)의 수는 미러 사본의 수에 따라 달라집니다. 또한 미러링을 사용하면 데이터 보호 및 시스템 가용성은 향상되지만 디스크 공간이 두 배 사용되거나 미러 사본이 있는 만큼 더 사용됩니다. 따라서 업무에 필수적인 휘발성 데이터에만 디스크 미러링을 사용해야 합니다.

미러 논리 볼륨은 같은 볼륨 그룹에 속해 있어야 합니다. 여러 볼륨 그룹에 걸쳐 미러링할 수는 없습니다.

이 절에서는 다음과 같은 내용을 다룹니다.

- “미러 쓰기 동작 제어” (38 페이지)
- “미러 논리 볼륨 동기화” (40 페이지)

기본 미러링 작업에 대한 자세한 내용은 1997년 Prentice Hall PTR에서 출판된 **Disk and File Management Tasks on HP-UX**를 참조하십시오.

미러 쓰기 동작 제어

미러 논리 확장 영역이 물리 확장 영역에 쓰여질 때는 **할당 정책**, 디스크 쓰기 **예약 정책** 및 크래시 복구 **동기화 정책**의 세 가지 정책이 적용됩니다. 이러한 정책은 HP SMH, `lvcreate` 명령 또는 `lvchange` 명령을 사용하여 설정할 수 있습니다.

할당 정책

미러 확장 영역은 **엄격** 또는 **비엄격**, **연속** 또는 **비연속** 정책으로 물리 볼륨에 할당할 수 있습니다. 기본적으로 미러 논리 볼륨의 할당 정책은 엄격, 비연속으로 설정됩니다.

엄격 및 비엄격 할당

엄격한 할당을 사용하려면 논리 확장 영역을 다른 물리 볼륨에 있는 물리 확장 영역에 미러링해야 합니다. 비엄격 할당을 사용하면 동일한 물리 볼륨에 있는 물리 확장 영역에 논리 확장 영역을 미러링할 수 있습니다. `lvcreate` 또는 `lvchange` 명령에 `-s y` 및 `-s n` 옵션을 사용하여 엄격 또는 비엄격 할당을 설정합니다.



주의: 비엄격 할당을 사용하면 논리 확장 영역이 동일한 디스크에 있는 서로 다른 물리 확장 영역에 미러링될 수 있으므로 LVM 미러링의 중복 효과가 감소될 수 있습니다. 따라서 이 디스크에서 장애가 발생하면 두 개의 데이터 사본을 모두 사용할 수 없게 됩니다.

연속 및 비연속 할당

연속 할당에는 세 가지 특징이 있습니다. 물리 확장 영역이 오름차순으로 할당되고, 미리 사본 내의 물리 확장 영역 사이에 간격이 없으며, 미리 사본의 모든 물리 확장 영역이 단일 물리 볼륨에 상주합니다. 비연속 할당을 사용하면 연속되지 않은 물리 확장 영역에 논리 확장 영역을 매핑할 수 있습니다. `lvcreate` 또는 `lvchange` 명령에 `-C y` 및 `-C n` 옵션을 사용하여 연속 또는 비연속 할당을 설정합니다.



참고: 루트 볼륨 그룹에서 할당된 논리 볼륨을 미러링하는 경우 연속 할당을 사용하여 각 논리 볼륨을 설정해야 합니다.

예약 정책

LVM 스케줄러는 논리 I/O 요청을 하나 이상의 물리 I/O 요청으로 변환한 후 하드웨어 수준에서 처리를 예약합니다. 예약은 미리 데이터와 미리되지 않은 데이터 모두에 대해 수행됩니다.

병렬과 순차라는 두 가지 I/O 예약 정책을 사용할 수 있습니다.

병렬 예약

병렬 예약 정책은 최대 I/O 성능을 얻을 수 있도록 미러링에 기본적으로 사용됩니다. 병렬 예약을 사용하면 미리 작업에서 모든 사본에 대해 동시에 씁니다. LVM은 해결되지 않은 I/O 작업이 가장 적은 물리 볼륨에서부터 데이터를 읽어 최적화된 방식으로 읽기 작업을 수행합니다. `lvcreate` 또는 `lvchange` 명령에 `-d p` 옵션을 사용하여 논리 볼륨의 예약 정책을 병렬로 설정합니다.

순차 예약

순차 예약 정책을 사용하면 미리 쓰기 작업이 순차적으로 진행됩니다. 즉, LVM은 한 미리 쓰기가 완료될 때까지 기다린 후에 다음 미리 쓰기를 시작합니다. 마찬가지로, LVM 미리 읽기도 미리 정의된 순서대로 진행됩니다. 미리의 일관성을 유지하기 위해 주의가 필요한 경우에만 순차 정책을 사용하는 것이 실용적입니다. `lvcreate` 또는 `lvchange` 명령에 `-d s` 옵션을 사용하여 논리 볼륨의 예약 정책을 순차로 설정합니다.

동기화 정책

미러 쓰기 캐시와 미러 일관성 복구라는 논리 볼륨의 두 가지 기능을 활성화하거나 비활성화하는 방법으로 미러 데이터의 일관성을 유지할 수 있습니다.

미러 쓰기 캐시를 사용한 동기화

MWC(Mirror Write Cache)를 사용하면 시스템 고장이나 장애가 발생한 후에 데이터를 빠르게 다시 동기화할 수 있지만, 일상적인 시스템 사용 시 성능이 저하됩니다.

MWC는 볼륨 그룹에서 I/O 쓰기가 발생하는 위치를 추적하고 이 활동을 디스크 상의 데이터 구조에 정기적으로 기록합니다. 물리 볼륨에 아직 기록되어 있지 않은 모든 미러 쓰기에 대해 추가 디스크 쓰기가 필요합니다. 따라서 런타임 I/O 쓰기 처리 속도가 느려지고 디스크에 임의로 액세스할 때 성능이 저하됩니다. 이미 기록된 디스크의 영역에 쓰는 경우에는 성능이 저하되지 않습니다. 장애로 중단된 후 시스템을 다시 부팅하면 운영 체제에서 MWC를 사용하여 일관성이 없는 데이터 블록을 신속하게 다시 동기화합니다.

데이터베이스 로그와 같이 순차적으로 액세스되는 논리 볼륨에서는 추가 디스크 쓰기가 많이 일어나지 않지만 액세스가 임의로 이루어지면 디스크 쓰기가 늘어납니다. 따라서 장애 복구 시간보다 런타임 성능이 더 중요한 경우에는 소수이거나 자주 쓰지 않는 큰 파일(256KB 이상)이 있는 데이터베이스 데이터나 파일 시스템이 포함된 논리 볼륨에서 MWC를 사용하지 말아야 합니다.

lvcreate 또는 lvchange 명령에 -M 옵션을 사용하여 MWC를 제어합니다.

미러 일관성 복구를 사용한 동기화

미러 일관성 복구를 사용하면 LVM으로 런타임 I/O 성능이 저하되지 않습니다. 하지만 시스템 장애가 발생한 후 볼륨 그룹이 다시 활성화되면 미러 일관성 복구를 사용하는 모든 논리 볼륨에서 전체 데이터 공간을 다시 동기화합니다. 동기화는 다시 부팅하고 액세스하는 작업을 방해하지 않고 백그라운드로 수행할 수 있지만 해당 작업 중에는 I/O 성능과 중복이 저하됩니다.

미러 일관성 메커니즘 없이 동기화

미러 일관성 복구를 사용하지 않는 경우 운영 체제의 런타임 동작은 앞에서 설명한 방법의 동작과 동일합니다. 그러나 장애가 발생한 후 LVM은 데이터를 다시 동기화하지 **않습니다**. 이 방법은 스왑 볼륨 그리고 트랜잭션 로그 파일과 같이 일관성 있는 데이터를 유지 또는 복구할 수단을 갖춘 응용 프로그램에 사용되는 볼륨(예: 데이터베이스)에서 유용합니다. 그러나 데이터베이스 로그 파일 자체는 MWC를 사용하는 미러 논리 볼륨으로 구성할 수 있습니다.

lvcreate 또는 lvchange 명령에 -c 옵션을 사용하여 미러 일관성 복구의 사용을 제어합니다.

미러 논리 볼륨 동기화

논리 볼륨의 미러 사본에 있는 데이터가 동기화되지 않거나 “못 쓰게 될” 수 있습니다. 예를 들어 디스크 정전으로 인해 LVM이 디스크에 액세스할 수 없는 경우 미러 데이터가 못 쓰게 됩니다. 이러한 상황에서는 미러 각 사본에서 동일한 데이터를 다시 만들기

위해 동기화를 수행해야 합니다. 일반적으로 동기화는 자동으로 수행되지만 수동으로 수행해야 하는 경우도 있습니다.

자동 동기화

현재 활성화되지 않은 볼륨 그룹을 부팅될 때 자동으로 또는 나중에 `vgchange` 명령을 사용하여 활성화하면 LVM은 미리 일관성 복구 정책을 활성화하여 모든 논리 볼륨의 미리 사본을 자동으로 동기화합니다. 또한 못 쓰게 된 것으로 표시된 물리 확장 영역의 데이터를 쓸 수 있는 확장 영역의 데이터로 교체합니다. 그렇지 않으면 자동 동기화가 수행되지 않고 수동 동기화가 필요하게 됩니다.

LVM은 다음의 경우에도 미리 데이터를 자동으로 동기화합니다.

- `lvmerge`의 `-m` 옵션을 사용하여 논리 볼륨의 미리 사본 수를 늘리면 새로 추가된 물리 확장 영역이 동기화됩니다.
- 정전 후 디스크가 다시 온라인 상태로 돌아온 경우

수동 동기화

`lvdisplay -v`를 사용하여 논리 볼륨의 상태를 보면 논리 볼륨에 못 쓰게 된 데이터가 들어 있는지 여부를 확인할 수 있습니다. 그런 다음 못 쓰게 된 물리 확장 영역이 들어 있는 디스크를 식별할 수 있습니다. `lvsync` 명령을 사용하여 하나 이상의 논리 볼륨에 있는 데이터를 수동으로 동기화하거나 `vgsync` 명령을 사용하여 하나 이상의 볼륨 그룹에 있는 모든 논리 볼륨의 데이터를 수동으로 동기화할 수 있습니다. 자세한 내용은 `lvdisplay(1M)`, `vgsync(1M)` 및 `lvsync(1M)`을 참조하십시오.

병렬 동기화

기본적으로 `lvsync` 명령은 순서대로 논리 볼륨을 동기화합니다. 즉, 한 번에 하나씩 명령줄에 지정된 논리 볼륨에서 작동하고 한 볼륨의 동기화가 완료될 때까지 기다린 후 다음 동기화를 시작합니다. HP-UX 11i v3의 2007년 9월 릴리즈부터 `-T` 옵션을 사용하여 논리 볼륨을 병렬로 동기화할 수 있습니다. `-T` 옵션을 사용하면 `lvsync` 명령으로 여러 스레드를 만들어 동일한 볼륨 그룹에 속하는 모든 논리 볼륨을 동시에 동기화함으로써 총 동기화 시간을 줄일 수 있습니다.



팁: `vgchange`, `lvmerge` 및 `lvextend` 명령은 못 쓰게 된 확장 영역의 자동 동기화를 억제하는 `-s` 옵션을 지원합니다. 여러 가지 미리 관련 작업을 수행하고 있는 경우 모든 작업이 완료될 때까지 확장 영역 동기화를 억제한 다음 `-T` 옵션과 함께 `lvsync`를 실행하여 모든 미리 볼륨을 병렬로 동기화할 수 있습니다. 예를 들어 `lvsync -T`와 함께 `vgchange -s`를 사용하여 미리 논리 볼륨 내의 볼륨 그룹에 대한 활성화 시간을 줄일 수 있습니다. 다른 예는 “부팅 디스크 미러링” (99 페이지)을 참조하십시오.

디스크 스페어링을 통해 디스크 중복 늘리기



참고: 버전 2.x 볼륨 그룹은 디스크 스페어링을 지원하지 않습니다.

디스크 스페어링을 사용하려면 선택적 제품인 HP MirrorDisk/UX가 필요합니다.

MirrorDisk/UX는 세 개 이상의 노드에 걸쳐 있는 고가용성 클러스터 내의 공유 LVM 환경에서 사용할 수 없습니다. 이러한 환경에서는 스페어링을 구성할 수 없습니다. 이런 경우 RAID 장치를 통해 하드웨어 미러링을 사용하는 것이 좋습니다. 하드웨어 미러링에서는 고유한 스페어링 형식을 지원할 수 있습니다.

미러 데이터가 있는 디스크가 장애를 일으키면 “손상된 디스크 교체” (126 페이지)에 설명된 대로 최대한 빨리 디스크를 교체해야 합니다. 미러 사본을 둘 이상 설정한 경우가 아니라면 디스크를 교체하기 전 논리 볼륨의 데이터에는 추가 미러 사본이 없습니다. 다중 미러링을 사용해도 하나의 미러 사본이 손실되므로 보안 수준은 저하됩니다.

이러한 문제가 발생하지 않도록 하려면 각 볼륨 그룹에서 하나 이상의 스페어 디스크를 사용하여 디스크 장애가 발생할 때 대체 장치로 사용해야 합니다. 이렇게 구성하면 사용자가 개입할 필요 없이 장애가 발생한 장치를 스페어 물리 볼륨으로 교체할 수 있도록 LVM이 볼륨 그룹을 자동으로 ‘재구성’합니다. 즉, 현재 장애가 발생한 디스크에 있는 모든 논리 볼륨의 데이터 사본이 대체 물리 볼륨에 만들어집니다. 이 프로세스를 자동 스페어링 또는 **스페어링**이라고 합니다. 스페어링 사용자가 논리 볼륨을 사용할 수 있는 상태에서 발생합니다. 그런 다음 사용자의 불편을 최소화하는 시점에 장애가 발생한 디스크를 교체하도록 예약할 수 있습니다. 이때 스페어 디스크의 데이터를 원래 디스크 또는 교체 디스크에 복사하고 스페어 디스크의 역할을 비어 있는 대기 디스크로 되돌립니다.

스페어링 기능을 수행하려면 다음 조건을 충족해야 합니다.

- 볼륨 그룹의 모든 논리 볼륨은 엄격한 미러링으로 구성되어야 하므로 미러 사본이 별도의 디스크에 유지 관리됩니다. 그 이유는 LVM이 결함 있는 디스크 자체가 아닌 손상되지 않은 디스크에서 스페어 디스크로 데이터를 복사하기 때문입니다.
- 적어도 하나의 물리 볼륨을 대기 스페어 볼륨으로 사용할 수 있어야 합니다. 이전 디스크 장애의 결과로 마지막 스페어 볼륨이 이미 사용 중인 경우에는 현재 사용 가능한 스페어 볼륨의 역할을 수행할 수 없습니다.
- 사용 가능한 스페어 디스크는 장애가 발생한 디스크보다 용량이 커야 합니다.

스페어 물리 볼륨의 디스크 공간은 디스크 장애 시 대체 디스크의 역할 이외의 다른 목적으로 확장 영역 할당에 사용할 수 없습니다. 따라서 해당 물리 확장 영역은 `pvdisplay`

및 `vgdisplay` 명령의 출력에서 `Total PE` 또는 `free PE`의 카운트에 포함되지 않습니다.



참고: 디스크 장애 시 적절한 성능을 유지 관리하는 것이 중요한 경우 각 버스에 스페어 물리 볼륨을 구성해야 합니다. 그러나 같은 버스에서 둘 이상의 디스크에 장애가 발생하는 경우에는 이 방법을 사용해도 성능이 저하됩니다.

`pvdisk` 및 `vgdisplay` 명령을 사용하면 지정된 물리 볼륨이 빈 대기 스페어 물리 볼륨인지 또는 현재 데이터를 갖고 있고 스페어 물리 볼륨으로 사용 중인지에 대한 정보 및 현재 사용할 수 없지만 해당 데이터가 스페어인 물리 볼륨에 대한 정보가 제공됩니다.

다중 경로 지정을 통해 하드웨어 경로 중복 늘리기

하드웨어가 같은 물리 볼륨에 대해 이중 케이블(이중 컨트롤러) 기능을 제공할 수 있습니다. 이러한 경우 같은 물리 볼륨에 대한 경로가 여러 개인 LVM을 구성할 수 있습니다. 기본 링크에 실패하면 대체 링크로 자동 전환됩니다. 다중 경로 지정을 사용하면 가용성이 높아집니다.



참고:

HP-UX 11i v3부터 대용량 저장소 스택에서는 LVM `pmlink`를 사용하지 않고 기본 다중 경로 지정을 지원합니다. 기본 다중 경로 지정에서는 LVM보다 많은 로드 밸런싱 알고리즘 및 경로 관리 옵션을 제공합니다. LVM의 대체 링크 대신 기본 다중 경로 지정을 사용하여 다중 경로 지정 장치를 관리하는 것이 좋습니다.

이전 버전과의 호환성을 위해서는 기존 `pmlinks`를 사용할 수 있습니다. 그러나 물리 볼륨에 기존 장치 특수 파일을 사용하고 `scsimgr` 명령을 사용하여 해당 기존 장치 특수 파일에 기본 다중 경로 지정을 비활성화해야 합니다. 자세한 내용은 <http://docs.hp.com>에 있는 `LVM Migration from Legacy to Agile Naming Model` 백서를 참조하십시오.

물리 볼륨에 대한 다중 경로 지정 설정

대체 링크를 사용하려면 `vgcreate`를 사용하여 볼륨 그룹을 만들어 기본 링크 및 대체 링크 장치 파일 이름을 지정할 수 있습니다. 두 링크 모두 같은 물리 볼륨에 대한 경로를 나타내야 합니다. 대체 링크에서는 `pvccreate`를 실행하지 마십시오. 기본 링크와 같은 물리 볼륨이어야 합니다. `vgcreate`를 사용하여 두 장치 파일 이름이 모두 동일한 디스크를 참조하고 있음을 나타내면 LVM에서는 첫 번째 이름을 기본 링크로 구성하고 두 번째를 대체 링크로 구성합니다.

예를 들어 디스크에 케이블 두 개가 있는데 하나를 기본 링크로 만들고 다른 하나를 대체 링크로 만들려면 다음 명령을 입력합니다.

```
# vgcreate /dev/vg01 /dev/dsk/c3t0d0 /dev/dsk/c5t0d0
```

볼륨 그룹의 일부인 물리 볼륨에 대체 링크를 추가하려면 `vgextend`를 사용하여 물리 볼륨에 새 링크를 지정합니다. 예를 들어 `/dev/dsk/c2t0d0`이 볼륨 그룹의 일부인데 물리 볼륨에 다른 연결을 추가하려면 다음 명령을 입력합니다.

```
# vgextend /dev/vg02 /dev/dsk/c4t0d0
```

기본 링크에 장애가 발생하면 LVM은 자동으로 기본 컨트롤러에서 대체 컨트롤러로 전환합니다. 그러나 `pvchange` 명령을 사용하여 언제든지 LVM이 다른 컨트롤러로 전환 되도록 할 수도 있습니다. 예를 들면 다음과 같습니다.

```
# pvchange -s /dev/dsk/c2t1d0
```

기본 링크가 복구된 후 다음과 같이 `pvchange`를 사용하여 대체 컨트롤러에서 원래 컨트롤러로 다시 전환되지 않도록 이전에 지시하지 않은 경우 LVM은 대체 컨트롤러에서 원래 컨트롤러로 자동으로 다시 전환합니다.

```
# pvchange -S n /dev/dsk/c2t2d0
```



참고: `pvchange`에 `-p` 옵션을 사용하면 자동 다시 전환을 비활성화하여 능동적 풀링을 비활성화할 수 있습니다. 자세한 내용은 `pvchange(1M)`를 참조하십시오.

물리 볼륨에 대한 현재 링크는 `vgdisplay`에 `-v` 옵션을 사용하여 볼 수 있습니다.

성능 계획

이 절에서는 LVM을 사용하여 최상을 성능을 얻을 수 있는 전략에 대해 설명합니다. 다음 내용에 대해 설명합니다.

- “일반 성능 요소” (44 페이지)
- “내부 성능 요소” (45 페이지)
- “디스크 스트라이핑을 통한 성능 향상” (46 페이지)
- “I/O 채널 분리를 통한 성능 향상” (48 페이지)

일반 성능 요소

다음은 전체 시스템 성능에 영향을 주지만 LVM 성능에는 영향을 주지 않을 수도 있는 요소입니다.

메모리 사용

LVM에서 사용되는 메모리의 크기는 볼륨 그룹을 만들 때 사용된 값과 열려 있는 논리 볼륨의 수에 따라 결정됩니다. LVM 메모리의 상당 부분이 확장 영역 매핑에 사용됩니다. 사용되는 메모리의 크기는 물리 볼륨의 최대 수에 각 볼륨 그룹에 대한 물리 볼륨 당 물리 확장 영역의 최대 수를 곱한 값에 비례합니다.

메모리 매개 변수와 관련하여 고려해야 할 다른 요소는 예상 시스템 성장 및 필요한 논리 볼륨 수입니다. 볼륨 그룹 최대 매개 변수를 오늘 시스템에 필요한 정확한 값으로 설정할 수 있습니다. 그러나 볼륨 그룹을 다른 디스크를 사용하여 확장하거나 한 디스크를 더 큰 디스크로 교체하려는 경우에는 `vgmodify` 명령을 사용해야 합니다.

CPU 사용

비LVM의 경우와 비교했을 때 유휴 시간을 관찰한 결과 시스템 CPU 사용에 큰 영향을 주는 심각한 경우는 발견되지 않았습니다.

LVM을 사용하여 CPU 사용에 영향을 주는, 구성 가능한 유일한 옵션인 미러 쓰기 일관성 캐시 작업을 수행하려면 별도의 CPU 주기가 필요합니다.

디스크 공간 사용

LVM은 자체 메타데이터용으로 각 물리 볼륨의 일부 디스크 공간을 예약합니다. 사용되는 공간의 크기는 볼륨 그룹을 만들 때 사용된 최대값에 비례합니다.

내부 성능 요소

다음 요소는 LVM을 통해 I/O의 성능에 직접 영향을 줍니다.

예약 정책

예약 정책은 미러링에서만 중요합니다. 미러링을 수행할 때 순차 예약 정책에는 미리 수에 비례하는 쓰기를 수행하는 데 필요한 시간이 필요합니다. 예를 들어 데이터 사본이 세 개인 논리 볼륨에서 순차 예약 정책을 사용하여 쓰기를 수행하면 병렬 정책을 사용하는 경우보다 세 배나 긴 시간이 걸립니다. 읽기 요청은 항상 한 장치로만 전달됩니다. 병렬 예약 정책에서 LVM은 각 읽기 요청을 가장 사용이 적은 장치로 전달합니다. 순차 예약 정책에서 LVM은 모든 읽기 요청을 `lvdisplay -v` 출력의 왼쪽에 표시된 장치로 전달합니다.

미러 쓰기 일관성 캐시

MWC(미러 쓰기 일관성 캐시)의 목적은 동기화되지 않은 미러 영역의 목록을 제공하는 것입니다. 볼륨 그룹이 활성화되어 있는 경우 LVM은 MWC에 있는 항목이 포함된 모든 영역을 상태가 양호한 사본 중 하나에서 나머지 모든 사본으로 복사합니다. 이 프로세스는 미러 간의 일관성을 유지하는 한편 데이터 품질은 보장하지 않습니다.

MWC를 사용하는 미러 논리 볼륨에 대한 쓰기 요청이 있을 때마다 LVM은 잠재적으로 한 번의 순차적 디스크 쓰기를 시작하여 MWC를 유지 관리합니다. 이러한 상황의 발생 여부는 액세스의 무작위 정도에 따라 다릅니다.

액세스의 무작위 정도가 많을수록 MWC의 누락 가능성도 높아집니다. MWC 항목을 가져오려면 항목을 사용할 수 있을 때까지 기다려야 할 수 있습니다. 현재 모든 MWC 항목을 진행 중인 I/O에서 사용하고 있는 경우 지정된 요청은 항목을 사용할 수 있을 때까지 요청 대기열에서 기다려야 할 수 있습니다.

미러링된 논리 볼륨의 또 한 가지 성능 고려 사항은 시스템 고장 후 미러 사본 간 불일치를 조정하는 방법입니다. MCR(Mirror Consistency Recovery)과 없음이라는 두 가지 재 동기화 방법을 사용할 수 있습니다. MWC를 사용할지 여부는 사용자의 컴퓨팅 환경에서 실행 시간과 복구 시간 중 시스템 성능의 어떤 면이 더 중요한가에 따라 달라집니다.

예를 들어 데이터베이스 시스템에서 미러링을 사용하는 고객의 경우는 데이터베이스 로깅 메커니즘에서 일관성 복구를 이미 제공하므로 데이터베이스 논리 볼륨에 대해 "없음"을 선택할 수 있습니다. 로그에 사용되는 논리 볼륨은 빠른 복구 시간이 중요한 경우 MWC를 사용하고 보다 뛰어난 런타임 성능이 중요한 경우에는 MCR을 사용합니다. 일반적으로 데이터베이스 로그는 하나의 프로세스에서 사용하며 순차적으로 액세스됩니다. 즉, 캐시는 거의 항상 사용되므로 MWC를 사용하는 경우 로그의 성능이 거의 저하되지 않습니다.

디스크 스페닝

여러 프로세스에서 집중적으로 사용하는 디스크 영역의 경우 이 디스크 영역의 데이터 공간을 최대한 많은 물리 볼륨에 분배하는 것이 좋습니다.

볼륨 그룹의 수

볼륨 그룹의 수는 MWC 문제와 직접적인 관련이 있습니다. 볼륨 그룹 당 MWC가 하나 뿐이기 때문에 MWC를 사용하는 동안 여러 개의 작은 무작위 쓰기 요청에 사용되는 디스크 공간은 가능한 한 별도의 볼륨 그룹에 두어야 합니다. 이 요소는 볼륨 그룹의 수와 관련된 결정에 영향을 주는 유일한 성능 고려 사항입니다.

물리 볼륨 그룹

이 요소는 여러 I/O 채널에서 서로 다른 여러 미러 사본을 분리하는 데 사용될 수 있습니다. 따라서 물리 볼륨 그룹을 정의해야 합니다. 이 요소를 사용하면 단일 지점 오류를 줄여 가용성을 높이고 하드웨어 수준에서 경합을 줄여 I/O 속도를 높일 수 있습니다.

예를 들어, 각 카드에 여러 디스크 장치가 있고 각 버스 변환기에 여러 카드가 있는 시스템에서 한 버스 변환기의 모든 디스크가 한 그룹에 들어가고 다른 버스 변환기의 모든 디스크가 다른 그룹에 들어가도록 물리 볼륨 그룹을 만듭니다. 이 구성을 사용하면 서로 다른 I/O 경로를 통해 액세스한 장치로 모든 미러를 만들 수 있습니다.

디스크 스트라이핑을 통한 성능 향상

디스크 스트라이핑은 논리적으로 연속된 데이터 블록(예: 동일한 파일의 여러 섹션)을 여러 디스크에 배포하므로 큰 파일을 순차적으로 읽고 쓸 때 I/O 속도가 빨라집니다(임의로 액세스할 때는 I/O 속도가 반드시 빨라지지는 않음).

디스크 스트라이핑의 단점은 파일이 둘 이상의 디스크에 분포되어 있으므로 한 디스크가 손실되면 많은 파일이 손상될 수 있다는 것입니다.

큰 파일을 저장할 때 해당 파일을 순차적으로 읽고 쓰는 경우가 일반적이고 I/O 성능이 중요하면 파일 시스템에서 디스크 스트라이핑을 사용하는 것을 고려하십시오.

디스크 스트라이핑을 사용하면 여러 디스크로 확장되는 논리 볼륨을 만들어 연속적인 데이터 블록을 서로 다른 디스크의 논리 확장 영역으로 이동할 수 있습니다. 예를 들어, 3-웨이 스트라이프 논리 볼륨 데이터가 디스크 세 개에 할당되어 있으며 세 디스크에 데이터 블록이 디스크 순서대로 돌아가면서 하나씩 저장되어 있다고 가정합니다. 각 디스크에는 데이터 블록의 1/3씩 저장됩니다. 이러한 블록의 크기를 논리 볼륨의 **스트라이프 크기**라고 합니다. 스트라이프 크기(K)는 버전 1.0 볼륨 그룹의 경우 4에서 32768 사이에 있는 2의 제곱수여야 하고, 버전 2.x 볼륨 그룹의 경우 4에서 262144 사이에 있는 2의 제곱수여야 합니다.

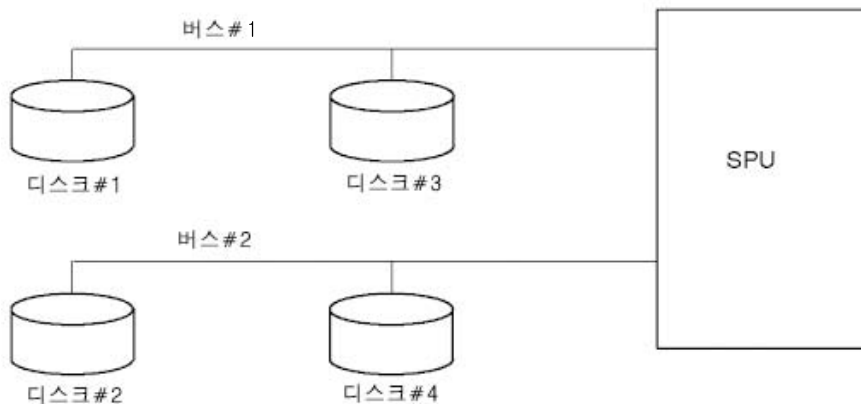
디스크 스트라이핑을 사용하면 **크기가 크고 순차적으로 액세스되는** 파일을 읽고 쓰는 응용 프로그램의 성능을 향상시킬 수 있습니다. 동시에 여러 디스크에 걸쳐 데이터가 액세스되므로 하나의 디스크에서 같은 작업이 수행될 때보다 필요한 시간이 줄어듭니다. 모든 스트라이프 디스크에 자신의 컨트롤러가 있으면 각각 동시에 데이터를 처리할 수 있습니다.

표준 명령을 사용하여 스트라이프 논리 볼륨을 관리할 수 있습니다. 예를 들어, `lvcreate`, `diskinfo`, `newfs`, `fsck` 및 `mount` 명령은 모두 스트라이프 논리 볼륨에서 작동합니다.

대부분 LVM 디스크를 사용할 경우에 적용되는 다음 지침은 성능상의 이유로 스트라이프 논리 볼륨에 적용됩니다.

- 유사한 디스크 간에 확장된 스트라이프 논리 볼륨을 사용하면 성능이 가장 좋아집니다. 디스크의 속도, 용량 및 인터페이스 종류를 가깝게 일치시킬수록 성능이 더 좋아집니다. 여러 가지 속도로 여러 디스크에 걸친 스트라이핑은 **가장 느린** 디스크에서보다 빠르지 않습니다.
- 디스크를 연결할 수 있는 인터페이스 카드 또는 버스가 둘 이상이면 이들 사이에 골고루 디스크를 분배합니다. 즉, 각 인터페이스 카드 또는 버스에 대략 같은 개수의 디스크가 연결되어야 합니다. 둘 이상의 버스를 사용하고 논리 볼륨의 스트라이프를 인터리브할 때 I/O 성능이 가장 좋게 됩니다. 예를 들어 버스가 두 개 있고 각 버스에 디스크가 두 개씩 있으면 **그림 2-2**에 설명된 것처럼 디스크 1을 버스 1에, 디스크 2를 버스 2에, 디스크 3을 버스 1에, 디스크 4를 버스 2에 정렬해야 합니다.

그림 2-2 버스 사이에 디스크 인터리빙



- 스트라이프 논리 볼륨에서 디스크를 결합하여 도달할 수 있는 최대 효율은 파일 시스템 자체 및 디스크가 연결된 버스의 최대 처리 효율로 제한되기 때문에 디스크의 수를 늘린다고 해서 항상 성능을 향상시킬 수 있는 것은 아닙니다.
- 디스크 스트라이핑은 사용자가 적고 전송이 크며 순차적으로 전송되는 응용 프로그램에서 매우 유용합니다. 데이터베이스처럼 동시에 무작위로 적은 횟수의 I/O가 수행되는 응용 프로그램은 디스크 스트라이핑을 통해 성능을 향상시킬 수 없습니다. 스트라이프 크기가 512바이트인 네 개의 디스크가 있는 경우 각 2K 요청이 모든 디스크로 전송됩니다. 2K 요청 하나는 디스크가 스트라이프되는 것과 거의 동시에 완료됩니다. 하지만 모든 디스크가 각 요청을 찾아야 하기 때문에 2K 요청 중 몇 개는 순차적으로 처리됩니다. 스트라이프되지 않은 시스템의 경우에는 각 디스크에서 병렬로 별도의 요청을 처리하기 때문에 실제로 성능이 더 좋을 수 있습니다.

최적 스트라이프 크기 결정

논리 볼륨의 스트라이프 크기는 스트라이프를 구성하는 각 데이터 블록의 크기와 같습니다. 스트라이프 크기를 버전 1.0 볼륨 그룹의 경우 4에서 32768 사이에 있는 2의 제곱수로 설정하고, 버전 2.x 볼륨 그룹의 경우 4에서 262144 사이에 있는 2의 제곱수로 설정할 수 있습니다. 기본값은 8192입니다.



참고: 논리 볼륨의 스트라이프 크기는 대개 512바이트인 디스크의 물리 섹터 크기와 관련이 없습니다.

스트라이프 논리 볼륨을 사용할 방식에 따라 지정할 스트라이프 크기가 결정됩니다. 최상의 결과를 위해 다음 지침을 따르십시오.

- HFS 파일 시스템에 스트라이프 논리 볼륨을 사용하려면 파일 시스템의 블록 크기를 가장 가깝게 반영하는 스트라이프 크기를 선택합니다. `newfs` 명령을 사용하면 파일 시스템을 만들 때 블록 크기를 지정하고 HFS에 8K의 기본 블록 크기를 제공할 수 있습니다.
- 스트라이프 논리 볼륨을 스왑 공간으로 사용하려면 최상의 성능을 위해 스트라이프 크기를 16K로 설정합니다. 스왑 구성에 대한 자세한 내용은 “스왑 논리 볼륨 관리” (111 페이지)를 참조하십시오.
- 스트라이프 논리 볼륨을 원시 데이터 파티션으로 사용하려는 경우(예: 장치를 직접 사용하는 데이터베이스 응용 프로그램의 경우) 스트라이프 크기는 응용 프로그램의 I/O 크기 이상이어야 합니다.

특정 상황에 대한 최적 스트라이프 크기를 결정하기 위해 실험이 필요할 수 있습니다. 스트라이프 크기를 변경하려면 논리 볼륨을 다시 만듭니다.

미러링과 스트라이핑 사이의 상호 작용

스트라이프 논리 볼륨을 미러링하면 스트라이프되지 않은 논리 볼륨의 경우와 같은 방식으로 읽기 I/O 성능이 향상됩니다. 단일 논리 확장 영역을 대상으로 하는 동시 읽기 I/O 요청은 하나가 아닌 두 개 또는 세 개의 서로 다른 물리 볼륨에서 처리합니다. 스트라이프 및 미러 논리 볼륨은 엄격한 할당 정책을 따릅니다. 즉, 데이터가 항상 다른 물리 볼륨에서 미러링됩니다.

I/O 채널 분리를 통한 성능 향상

I/O 채널 분리는 별도의 HBA(호스트 버스 어댑터)와 케이블을 사용해 액세스한 LVM 디스크에 데이터의 미러 사본이 상주할 것을 요구하는 LVM 구성에 대한 접근 방법입니다. I/O 채널 분리를 사용하면 하드웨어의 가능한 단일 지점 오류 수를 줄여 가용성을 높이고 성능을 향상시킬 수 있습니다. 한 카드를 통해 별도의 두 디스크에 데이터를 미러링하면 카드에 장애가 발생하는 경우 시스템에서 오류가 발생할 수 있습니다.

여러 HBA에 디스크를 미러링하여 HBA가 여러 개이며 단일 버스가 있는 시스템에서 I/O 채널을 분리할 수 있습니다. 별도의 물리 볼륨 그룹에 논리 확장 영역을 미러링해야 하는 PVG **엄격** 할당이라는 정책을 설정하면 채널 분리를 더욱 보장할 수 있습니다. **물리 볼륨 그룹**은 볼륨 그룹 내에 있는 물리 볼륨의 하위 그룹입니다.

ASCII 파일인 `/etc/lvm/pvg`에는 물리 볼륨 그룹의 매핑 정보가 모두 포함되어 있지만 매핑은 디스크에 기록되지 않습니다. 물리 볼륨 그룹에는 고정된 명명 규칙이 없습니다. 여기에는 PVG0, PVG1 등의 이름을 지정할 수 있습니다. `/etc/lvm/pvg` 파일은 `vgcreate`, `vgextend` 및 `vgreduce` 명령을 사용하여 만들고 업데이트하지만 텍스트 편집기를 사용하여 파일을 편집할 수 있습니다.

I/O 채널 분리는 가용성을 높여(LVM은 가장 액세스 가능한 논리 확장 영역에서 데이터 읽기에 유연성을 제공) 성능을 향상시키므로 데이터베이스에 유용합니다. 물리 볼륨 그룹을 정의하여 I/O 장치를 확장하면 한 HBA에서 장애가 발생한 경우에도 데이터 손실을 방지할 수 있습니다.

물리 볼륨 그룹을 사용하는 경우에는 논리 볼륨에 PVG 엄격 할당 정책을 사용하는 것을 고려하십시오.

복구 계획

LVM의 주된 장점 중 하나인 구성의 유연성 때문에 복구 중에 문제가 발생할 수도 있습니다. 다음은 복구 시간을 최소화하는 구성을 만드는 데 도움이 되는 지침입니다.

- 루트 볼륨 그룹에 있는 디스크의 수를 최소한으로 유지하십시오. 루트 볼륨 그룹을 미리랑하는 경우에도 세 개의 디스크를 사용하는 것이 좋습니다.

루트 볼륨 그룹에 디스크가 많으면 루트 볼륨 그룹 내에 있는 보조 디스크의 LVM 구성을 복구하는 작업이 복잡해지기 때문에 다시 설치하기가 어려워집니다.

작은 루트 볼륨 그룹은 빠르게 복구됩니다. 경우에 따라 최소한의 시스템을 다시 설치하고 백업을 복원하며 세 시간 이내에 진단과 하드웨어 교체를 마친 후 다시 온라인으로 돌아올 수도 있습니다. 다른 장점으로는 이전 루트 디스크 레이아웃과 완전히 일치할 필요가 없다는 것도 있습니다.

쿼럼 제한 때문에 루트 볼륨 그룹에 디스크가 두 개 있는 것보다는 세 개 있는 것이 좋습니다. 디스크가 두 개인 루트 볼륨 그룹에서 디스크 하나가 손실되면 쿼럼을 재정의하여 볼륨 그룹을 활성화해야 할 수도 있습니다. 다시 부팅하여 디스크를 교체해야 하는 경우 쿼럼을 다시 정의하려면 부팅 과정을 중단해야 합니다. 볼륨 그룹에 디스크가 세 개 있고 이 디스크가 서로 분리되어 있어 한 번의 하드웨어 장애로 한 디스크만 영향을 받는 경우 한 디스크에만 장애가 발생하면 시스템에서 쿼럼을 유지할 수 있습니다.

루트 볼륨 그룹을 최소 크기보다 크게 확장하는 이유는 다음 두 가지입니다.

- 매우 작은 루트 디스크

이 경우 더 큰 디스크로 마이그레이션하거나 설치하는 것이 좋습니다.

- 대형 메모리 시스템용 스왑에 대한 덤프 제공

덤프 대상으로 지정된 스왑 볼륨은 루트 볼륨 그룹에 있어야 합니다. 더 나은 해결 방법은 덤프 전용 별도의 디스크를 구성하는 것입니다.

- Omniback 또는 Networker와 같은 네트워크 기반 백업 프로그램을 기본 루트 볼륨 그룹 백업에 사용하지 마십시오. 이런 유틸리티는 복잡하기 때문에 처리 재개가 상당히 지연될 수 있습니다.

Ignite-UX와 `make_net_recovery` 명령을 사용하여 루트 볼륨 그룹을 백업 및 복구하는 것이 좋습니다.

- 적절한 설명서를 만듭니다.

`ioscan -kf`, 모든 그룹에 대한 `vgcfgrestore -lv` 및 `vgscan -pv`의 출력과 `lvlnboot -v`는 최소 요구 사항입니다. 이러한 최소 요구 사항과 모든 그룹에 대한 `vgdisplay -v` 및 모든 볼륨에 대한 `lvdisplay -v`의 출력을 사용할 수 있으면 거의 모든 문제로부터 복구할 수 있습니다. 확장 영역 매핑은 헤더가 손상된 LVM 볼륨의 복구에 중요합니다. 또한 모든 물리 볼륨의 `pvdisplay -v` 출력은 그만큼 중요하지는 않지만 자세한 볼륨 그룹 정보를 제공합니다. 인쇄본이 꼭 필요하거나 실제로 도움이 되지 않을 수도 있지만 복구 중에 액세스 가능성은 중요하며 적절한 계획이 필요합니다.

- 보조 볼륨 그룹을 구성할 때 소수의 대형 그룹보다는 다수의 소형 그룹을 만드는 것이 더 나을 것입니다.

루트 디스크를 다시 구축한 후에 그룹에서 디스크 하나가 누락되었기 때문에 수십 GB의 데이터를 다시 로드한다는 것은 어려울 수 있습니다. 또한 한 그룹 내의 치명적인 단일 디스크 장애 범위도 많은 작은 그룹으로 최소화됩니다.

크기가 커지면 복잡해집니다. 단일 그룹에 있는 디스크의 수가 많을수록 관리자가 그룹 전체에 영향을 주는 오류를 일으킬 가능성도 커집니다. 필요한 경우 더 작은 그룹을 확인하고 가져오는 것이 좀더 간단합니다. 필요한 경우 더 작은 그룹을 개념화하고 매핑하는 것도 더 간단합니다.

LVM 시스템 복구 준비

시스템 장애가 발생한 경우에 시스템 데이터와 구성을 복구할 수 있도록 하려면 다음 단계를 수행하십시오.

1. LVM의 패치를 모두 로드합니다.
2. Ignite-UX를 사용하여 루트 볼륨 그룹의 복구 이미지를 만듭니다. Ignite-UX는 모든 시스템 데이터를 백업하기 위한 것이 아니지만 이 프로그램을 다른 데이터 복구 응용 프로그램과 함께 사용하여 시스템 전체를 복구할 방법을 마련할 수 있습니다.
3. 시스템에서 다른 중요한 데이터의 정기 백업을 수행합니다.
유효한 백업이 없으면 데이터의 일부 또는 전부를 잃게 될 위험이 있습니다.
4. 시스템 구성을 정기적으로 인쇄합니다.

복구 중에는 시스템에 저장된 구성 세부 사항에 액세스할 수 없을 수 있습니다. 인쇄된 사본은 귀중한 참조가 됩니다. 구성 세부 사항은 한 주에 한 번 **그리고** 변경 사항이 있을 때마다 인쇄하는 것이 좋습니다. 명령 중 일부는 대량의 출력을 만들어 냅니다. 인쇄하는 대신 정보를 파일로 출력한 다음 파일을 테이프에 저장해 두면 펄

요한 경우에 정보를 빨리 복구할 수 있습니다. 이 구성 파일을 3단계의 백업 과정에 포함할 수도 있습니다.

cron 작업이 정기적으로 실행되도록 설정하여 시스템에서 자동으로 백업하도록 하면 구성을 가장 쉽게 저장할 수 있습니다.

다음 명령을 사용하여 유용한 출력을 얻습니다.

```
/usr/sbin/ioscan -fk
/usr/sbin/vgdisplay -v
/usr/sbin/lvlnboot -v
/usr/sbin/lvdisplay -v /dev/vgXX/lvYY(모든 논리 볼륨에 대해)
/usr/sbin/pvdisplay -v /dev/dsk/c#t#d0(모든 LVM 디스크에 대해)
lp /etc/fstab
```

대안으로 구성의 변경 사항을 모두 감지하여 변경된 내용만 인쇄하는 지능적인 스크립트를 작성할 수 있습니다. 이 절의 끝에는 스크립트 예가 포함되어 있습니다.

5. 각 구성이 변경된 후에 LVM 구성을 백업합니다.

vgcfsbackup 명령은 디스크의 시스템 영역에서 LVM 헤더를 디스크 파일로 복사하며 이 파일은 /etc/lvmconf 디렉토리에 있습니다. 이 정보가 디스크 파일에 저장되면 파일 시스템을 백업하는 동안 테이프에 백업할 수 있습니다.

이 파일에 있는 정보를 사용하면 디스크가 교체되거나 LVM 구성이 손상된 경우에 디스크에 있는 LVM 헤더를 교체할 수 있습니다.

LVM 구성의 일부를 변경할 때마다 이러한 구성 백업을 마련해 두는 것이 중요합니다. 기본적으로 모든 명령에서 백업을 수행하므로 각 명령을 수행한 후에 수동으로 vgcfsbackup을 실행할 필요가 없습니다.

이 작업은 변경 여부에 관계 없이 정기적으로 수행합니다. 다음 명령을 입력합니다.

```
# /usr/sbin/vgcfsbackup /dev/vgXX (for every volume group)
```

6. 루트 볼륨 그룹에서 변경 사항이 생길 때마다 부팅 구조를 업데이트합니다.

이 작업은 부팅 디스크에서 LVM을 사용하는 경우에만 필요합니다. 루트 볼륨 그룹을 변경할 때마다(보통은 이름이 /dev/vg00) 부팅 디스크에 있는 BDRA를 업데이트해야 합니다. 이 작업은 보통 LVM 명령에 의해 자동으로 수행됩니다. BDRA를 수동으로 업데이트하려면 다음 명령을 입력합니다.

```
# /usr/sbin/lvlnboot -R
```

LVM 구성 기록의 예제 스크립트

다음은 현재 LVM 및 I/O 구성을 캡처하는 스크립트입니다. 현재 구성이 이전에 캡처한 구성과 다를 경우 스크립트는 업데이트된 구성 파일을 인쇄하고 시스템 관리자에게 알립니다.

```
#!/usr/bin/ksh
WORKDIR=/lvmbakup # directory is regularly backed up
LOG=$WORKDIR/log
```

```

SYSADM=root
if [ -f "$LOG" ]
then
    rm -f "$LOG"
fi
if [ ! -d "$WORKDIR" ]
then
    echo "missing directory $WORKDIR" exit 1
fi
cd $WORKDIR
/usr/sbin/vgdisplay -v -F > vgdisplay.new
LVMVGS=`grep vg_name vgdisplay.new | cut -d: -f1 | cut -d= -f2`
LVMPVOLS=`grep pv_name vgdisplay.new | cut -d: -f1 | cut -d= -f2 | cut -d, -f1`
LVMLVOLS=`grep lv_name vgdisplay.new | cut -d: -f1 | cut -d= -f2`
/usr/sbin/pvdisplay -v $LVMPVOLS > pvdisplay.new
/usr/sbin/lvdisplay -v $LVMLVOLS > lvdisplay.new
/usr/sbin/lvlnboot -v > lvlnboot.new 2> /dev/null
/usr/sbin/ioscan -fk > ioscan.new
cp /etc/fstab fstab.new
for CURRENT in *new.
do
    ORIG=${CURRENT%.new}
    if diff $CURRENT $ORIG > /dev/null then
        # files are the same....do nothing
        rm $CURRENT
    else
        # files differ...make the new file the current file, move old
        # one to file.old.
        echo `date` "The config for $ORIG has changed." >> $LOG
        echo "Copy of the new $ORIG config has been printed" >> $LOG
        lp $CURRENT
        mv $ORIG ${ORIG}.old.
        mv $CURRENT $ORIG
    fi
done
if [ -s "$LOG" ]
then
    mailx -s "LVM configs have changed" $SYSADM < $LOG
fi
exit 0

```

3 LVM 관리

이 절에서는 LVM의 일별 작업에 대해 설명합니다. 다음 내용에 대해 설명합니다.

- “관리 도구” (53 페이지)
- “LVM 정보 표시” (56 페이지)
- “일반적인 LVM 작업” (58 페이지)
- “디스크 이동 및 재구성” (81 페이지)
- “파일 시스템 논리 볼륨 관리” (105 페이지)
- “스왑 논리 볼륨 관리” (111 페이지)
- “덤프 논리 볼륨 관리” (112 페이지)
- “하드웨어 문제” (112 페이지)

관리 도구

HP-UX에서는 LVM 구성을 관리하는 두 가지 도구를 제공합니다.

- **HP SMH(HP System Management Homepage):** 대부분의 LVM 작업을 수행하는 데 쉽게 사용할 수 있는 GUI를 제공하는 HP-UX 도구입니다. HP SMH를 사용하면 관리 명령에 대해 자세히 알아야 할 필요가 최소화되거나 없게 지므로 시간을 절약할 수 있습니다. 가능하면 HP SMH를 사용하여 LVM 구성을 관리하고 특히 새 작업을 수행하는 경우에는 HP SMH를 사용하십시오. HP SMH를 사용하면 다음 작업을 수행할 수 있습니다.
 - 볼륨 그룹 만들기 또는 제거
 - 볼륨 그룹에서 디스크 추가 또는 제거
 - 볼륨 그룹 활성화 및 비활성화
 - 볼륨 그룹 내보내기 및 가져오기
 - 논리 볼륨 만들기, 제거 또는 수정
 - 논리 볼륨 크기 늘리기
 - 논리 볼륨에서 파일 시스템 만들기 또는 크기 늘리기
 - 미러 논리 볼륨 만들기 및 수정
 - 스트라이프 논리 볼륨 만들기
 - 미러 논리 볼륨 분리 및 미러 사본 병합
 - 미러 논리 볼륨 내에서 미러 사본 추가 및 제거
 - 물리 볼륨 그룹 만들기 및 수정

HP SMH를 사용하려면 `/usr/sbin/smh` 명령을 입력합니다.

HP SMH 사용에 대한 도움말은 HP SMH 온라인 도움말을 참조하십시오.

- **LVM 명령줄 인터페이스:** “물리 볼륨 관리 명령” (54 페이지), “볼륨 그룹 관리 명령” (54 페이지) 및 “논리 볼륨 관리 명령” (55 페이지)에서 설명한 대로 LVM에는 LVM 작업을 수행하기 위한 다양한 하위 수준 사용자 명령이 있습니다.

다음 표에서는 지정된 작업을 수행하는 명령에 대한 개요를 제공합니다. 자세한 내용은 HP-UX Reference를 참조하십시오.

표 3-1 물리 볼륨 관리 명령

작업	명령
볼륨 그룹에서 물리 볼륨의 특성 변경	pvchange
볼륨 그룹에서 사용할 물리 볼륨 만들기	pvcreate
볼륨 그룹의 물리 볼륨에 대한 정보 표시	pvdisplay
한 물리 볼륨에서 다른 물리 볼륨으로 데이터 이동	pvmove
LVM 제어에서 물리 볼륨 제거	pvremove
물리 볼륨 검사 또는 복구	pvck*
디스크 볼륨이 LVM 제어 하에 있는지 확인	lvmchk

표 3-2 볼륨 그룹 관리 명령

작업	명령
볼륨 그룹 만들기	vgcreate
볼륨 그룹 제거	vgremove
볼륨 그룹의 특성 활성화, 비활성화 또는 변경	vgchange
볼륨 그룹의 구성 매개 변수 수정, 물리 볼륨 크기 변경 처리	vgmodify*
볼륨 그룹 구성 정보 백업	vgcfgbackup
구성 파일의 볼륨 그룹 구성 복원	vgcfgrestore
볼륨 그룹 정보 표시	vgdisplay
볼륨 그룹 및 관련 논리 볼륨 내보내기	vgexport
시스템으로 볼륨 그룹 가져오기, 기존 볼륨 그룹을 LVM 구성 파일에 다시 추가	vgimport
전체 물리 볼륨에서 논리 볼륨 및 볼륨 그룹 검색, LVM 구성 파일 복구	vgscan
볼륨 그룹에 디스크 추가	vgextend
볼륨 그룹에서 디스크 제거	vgreduce
볼륨 그룹에서 미리 논리 볼륨 동기화	vgsync

표 3-2 볼륨 그룹 관리 명령 (계속)

작업	명령
물리 볼륨의 볼륨 그룹 ID 수정	vgchgid
기존 장치 파일에서 영구 장치 파일로 볼륨 그룹 마이그레이션	vgdsf
볼륨 그룹 버전과 관련된 제한 표시	lvmdm

표 3-3 논리 볼륨 관리 명령

작업	명령
논리 볼륨 만들기	lvcreate
논리 볼륨 수정	lvchange
논리 볼륨 정보 표시	lvdisplay
디스크 공간을 할당하여 논리 볼륨 크기 늘리기	lvextend
논리 볼륨의 크기 줄이기	lvreduce
볼륨 그룹에서 하나 이상의 논리 볼륨에 대한 디스크 공간 할당 제거	lvremove
논리 볼륨을 루트, 기본 스왑 또는 덤프 볼륨으로 준비, 부팅 물리 볼륨에서 부팅 정보 업데이트	lvlnboot *
논리 볼륨을 루트, 기본 스왑 또는 덤프 볼륨으로 만드는 링크 제거	lvrmboot *
미러 논리 볼륨을 두 논리 볼륨으로 분리	lvsplit
두 논리 볼륨을 하나의 미러 논리 볼륨으로 병합	lvmerge
미러 논리 볼륨에서 미러 사본 동기화	lvsync

* pvck, vgmodify, lvlnboot 및 lvrmboot 명령은 버전 2.x 볼륨 그룹에서 지원되지 않습니다.

명령줄 인터페이스는 HP SMH보다 강력한 반면 위험하므로 HP SMH를 통해 사용할 수 없는 옵션을 제공할 수 있습니다. 예를 들어, 다음 작업은 현재 HP SMH를 통해 수행할 수 없습니다. 이러한 작업에 대해서는 HP-UX 명령을 사용하십시오.

- “스페어 디스크 만들기” (86 페이지)
- “스페어 디스크 복귀” (86 페이지)
- “미러 논리 볼륨 동기화” (40 페이지)
- “다른 물리 볼륨으로 데이터 이동” (84 페이지)
- “손상된 디스크 교체” (126 페이지)

이 장의 나머지 부분에서는 HP-UX 명령을 사용하여 LVM 작업을 수행하는 방법에 대해 설명합니다. 그러나 HP SMH는 대부분의 관리 작업에서 선택할 수 있는 도구입니다.

LVM 정보 표시

볼륨 그룹, 논리 볼륨 또는 물리 볼륨에 대한 정보를 표시하려면 세 가지 명령 중 하나를 사용합니다. 각 명령은 자세한 출력을 표시하는 -v 옵션과 스크립팅에 도움이 되는 -F 옵션을 지원합니다.

볼륨 그룹 정보

vgdisplay 명령을 사용하여 볼륨 그룹에 대한 정보를 표시합니다. 예를 들면 다음과 같습니다.

```
# vgdisplay -v vg01
-- 볼륨 그룹 --
VG 이름                /dev/vg01
VG 쓰기 권한          읽기/쓰기
VG 상태              사용 가능
최대 LV                255
현재 LV                1
열린 LV               1
최대 PV                16
열린 PV               1
활성 PV               1
PV당 최대 PE          1016
VGDA                  2
PE 크기(MB)           4
총 PE                 508
할당된 PE             508
사용 가능한 PE        0
총 PVG                0
총 스페어 PV          0
사용 중인 총 스페어 PV 0
VG 버전              1.0
VG 최대 크기          1082g
VG 최대 확장 영역      69248

-- 논리 볼륨 --
LV 이름                /dev/vg01/lvol1
LV 상태              사용 가능/동기화됨
LV 크기(MB)           2032
현재 LE              125
할당된 PE            508
사용된 PV             1

-- 물리 볼륨 --
PV 이름                /dev/disk/disk42
PV 상태              사용 가능
총 PE                 508
사용 가능한 PE        0
자동 전환             켜짐
능동적 폴링           켜짐
```


vgdisplay 명령을 사용하여 메모리의 LVM 구성에 문제가 있는지 여부를 확인합니다. LVM 구성이 제대로 작동하고 있는 경우에는 오류 메시지가 표시되지 않으며 다음과 같이 표시됩니다.

- 상태는 사용 가능(또는 Serviceguard 볼륨 그룹의 경우 사용 가능/제외)입니다.
- 모든 물리 볼륨이 활성화되어 있습니다. 즉, 현재 물리 볼륨 수(현재 PV)가 활성화 물리 볼륨 수(활성 PV)와 동일합니다.
- 모든 논리 볼륨이 열려 있습니다. 즉, 현재 LV가 열린 LV와 동일합니다.

물리 볼륨 정보

pvdisplay 명령을 사용하여 물리 볼륨에 대한 정보를 표시합니다. 예를 들면 다음과 같습니다.

```
# pvdisplay -v /dev/disk/disk47
-- 물리 볼륨 --
PV 이름                /dev/disk/disk47
VG 이름                /dev/vg00
PV 상태              사용 가능
할당 가능             예
VGDA                  2
현재 LV               9
PE 크기(MB)          4
총 PE                 1023
사용 가능 PE          494
할당된 PE             529
오래된 PE             0
IO 시간 제한(초)      기본값
자동 전환             켜짐
능동적 폴링           켜짐

-- 물리 볼륨 배포 --
LV 이름                LV의 LE    LV에 대한 PE
/dev/vg00/lvol1        25         25
/dev/vg00/lvol2        25         25
/dev/vg00/lvol3        50         50

--- 물리 확장 영역 ---
PE   상태   LV                LE
0000 현재   /dev/vg00/lvol1    0000
0001 현재   /dev/vg00/lvol1    0001
0002 현재   /dev/vg00/lvol1    0002
...
1021 사용 가능                0000
1022 사용 가능                0000
```

물리 볼륨이 제대로 작동하고 있는 경우 오래된 PE는 0입니다.

논리 볼륨 정보

lvdisplay 명령을 사용하여 논리 볼륨에 대한 정보를 표시합니다. 예를 들면 다음과 같습니다.

```
# lvsdisplay -v /dev/vg00/lvol1
-- 논리 볼륨 --
LV 이름 /dev/vg00/lvol1
VG 이름 /dev/vg00
LV 권한 읽기/쓰기
LV 상태 사용 가능/동기화됨
미러 복사본 0
일관성 복구 MWC
일정 병렬
LV 크기(MB) 100
현재 LE 25
할당된 PE 25
스트라이프 0
스트라이프 크기(KB) 0
손상된 블록 깨짐
할당 엄격/연속
IO 시간 제한(초) 기본값

-- 논리 볼륨 배포 --
PV 이름 PV의 LE PV의 PE
/dev/disk/disk42 25 25

-- 논리 확장 영역 --
LE PV1 PE1 상태 1
0000 /dev/disk/disk42 0000 현재
0001 /dev/disk/disk42 0001 현재
0002 /dev/disk/disk42 0002 현재
```

일반적인 LVM 작업

이 절에서는 다음 내용에 대해 설명합니다.

- “LVM 사용을 위한 디스크 초기화” (59 페이지)
- “볼륨 그룹 만들기” (60 페이지)
- “볼륨 그룹에 디스크 추가” (62 페이지)
- “볼륨 그룹에서 디스크 제거” (62 페이지)
- “논리 볼륨 만들기” (63 페이지)
- “논리 볼륨 확장” (65 페이지)
- “논리 볼륨 축소” (66 페이지)
- “논리 볼륨에 미러 추가” (67 페이지)
- “논리 볼륨에서 미러 제거” (68 페이지)
- “논리 볼륨 이름 바꾸기” (68 페이지)
- “논리 볼륨 제거” (69 페이지)
- “볼륨 그룹 내보내기”
- “볼륨 그룹 가져오기” (70 페이지)
- “볼륨 그룹 매개 변수 수정” (70 페이지)
- “볼륨 그룹 정지 및 다시 시작” (75 페이지)

- “볼륨 그룹 이름 바꾸기” (76 페이지)
- “볼륨 그룹 분리” (77 페이지)
- “볼륨 그룹 제거” (78 페이지)
- “미러 논리 볼륨 백업” (79 페이지)
- “볼륨 그룹 구성 백업 및 복원” (80 페이지)

LVM 사용을 위한 디스크 초기화



주의: `pvcreate`를 사용하여 디스크를 초기화하면 현재 디스크에 있는 기존 데이터가 손실됩니다.



참고: 디스크가 시스템에 이미 연결되어 있으면 이 절차의 처음 네 단계를 건너뛰니다.

물리 볼륨으로 사용하도록 디스크를 초기화하려면 다음 단계를 수행합니다.

1. 시스템을 종료하고 시스템의 전원을 끕니다.
2. 시스템에 디스크를 연결하고 전원을 켭니다. 특정 유형의 디스크를 추가하는 방법에 대한 자세한 내용 및 지침은 해당 장치 설명서를 참조하십시오.
3. 디스크의 전원을 켭니다.
4. 시스템을 부팅합니다.
5. 디스크의 관련 장치 파일을 확인합니다. 시스템에 연결된 디스크와 해당 장치 파일 이름을 표시하려면 `ioscan` 명령과 `-f`, `-N` 및 `-n` 옵션을 함께 사용합니다. 예를 들면 다음과 같습니다.

```
# ioscan -f -n -N -C disk
```

자세한 내용은 `ioscan(1M)`을 참조하십시오.

6. `pvcreate` 명령을 사용하여 디스크를 물리 볼륨으로 초기화합니다. 예를 들면 다음과 같습니다.

```
# pvcreate /dev/rdisk/disk3
```

디스크에 특성 장치 파일을 사용합니다.

부팅 장치로 사용할 디스크를 초기화하고 있는 경우 `pvcreate`에 `-B` 옵션을 추가하여 LIF 볼륨 및 부팅 유틸리티용으로 디스크의 영역을 예약합니다. HP Integrity 서버에 부팅 디스크를 만들 경우 장치 파일에 HP-UX 파티션 번호(2)가 지정되어 있는지 확인합니다. 예를 들면 다음과 같습니다.

```
# pvcreate -B /dev/rdisk/disk3_p2
```



참고: 버전 2.x 볼륨 그룹은 부팅 가능한 물리 볼륨을 지원하지 않습니다. 버전 2.x 볼륨 그룹에서 디스크를 사용할 경우에는 `-B` 옵션을 사용하지 마십시오.

디스크가 초기화된 후에는 물리 볼륨이라고 합니다.

볼륨 그룹 만들기

볼륨 그룹을 만들려면 `vgcreate` 명령을 사용합니다. 옵션은 버전 1.0 볼륨 그룹을 만드는지, 버전 2.x 볼륨 그룹을 만드는지 여부에 따라 달라집니다.

볼륨 그룹 장치 파일 만들기

HP-UX 11i v3의 2008년 3월 릴리즈부터 `vgcreate` 명령은 볼륨 그룹 버전에 관계없이 `/dev/vgname/group` 장치 파일을 자동으로 만들어 볼륨 그룹을 관리합니다. 2008년 3월 이전의 HP-UX 릴리즈를 사용하고 있거나 그룹 파일의 보조 번호를 지정하려는 경우에는 `vgcreate` 명령을 실행하기 전에 `/dev/vgname/group`을 만들어야 합니다. 그룹 파일이 없는 경우 `vgcreate`로 만들 수 없으면 `vgcreate`에 다음 메시지가 표시됩니다.

`vgcreate: "/dev/vgname/group": 문자 장치가 아닙니다.`

볼륨 그룹 장치 파일을 만들려면 다음 단계를 수행합니다.

1. 볼륨 그룹에 대한 디렉토리를 만듭니다. 예를 들면 다음과 같습니다.

```
# mkdir /dev/vgname
```

기본적으로 `vgname`은 `vgnn`입니다. 여기서, `nn`은 전체 볼륨 그룹 내에서 고유한 숫자입니다. 그러나 고유 이름을 최대 255자까지 선택할 수 있습니다.

2. `mknod` 명령을 사용하여 볼륨 그룹 디렉토리에 `group`이라는 장치 파일을 만듭니다. 예를 들면 다음과 같습니다.

```
# mknod /dev/vgname/group c major 0xminor
```

장치 파일 이름 다음의 `c`는 `group`이 특성 장치 파일이라는 것을 지정합니다.

`major`는 그룹 장치 파일의 주 번호입니다. 버전 1.0 볼륨 그룹의 경우는 64이고, 버전 2.x 볼륨 그룹의 경우는 128입니다.

`minor`는 그룹 파일의 보조 번호로 16진수입니다. 버전 1.0 볼륨 그룹의 경우 `minor`는 `0xnn0000` 형식입니다. 여기에서 `nn`은 모든 버전 1.0 볼륨 그룹에서 고유한 번호입니다. 버전 2.x 볼륨 그룹의 경우 `minor`는 `0xnnn000` 형식입니다. 여기에서 `nnn`은 모든 버전 2.x 볼륨 그룹에서 고유한 번호입니다.

`mknod`에 대한 자세한 내용은 `mknod(1M)`를 참조하십시오. 주 번호 및 보조 번호에 대한 자세한 내용은 “장치 번호 형식” (27 페이지)을 참조하십시오.

버전 1.0 볼륨 그룹 만들기

버전 1.0 볼륨 그룹을 만들려면 `vgcreate` 명령을 사용하여 포함할 각 물리 볼륨을 지정합니다. 예를 들면 다음과 같습니다.

```
# vgcreate /dev/vgname /dev/disk/disk3
```

블록 장치 파일을 사용하여 각 디스크를 볼륨 그룹에 포함합니다. 하나의 명령으로 볼륨 그룹에 모든 물리 볼륨을 할당하거나 단일 물리 볼륨을 사용하여 볼륨 그룹을 만들 수 있습니다. 물리 볼륨은 기존 볼륨 그룹의 일부일 수 없습니다.

다음 옵션을 사용하여 볼륨 그룹 속성을 설정할 수 있습니다.

-V 1.0	버전 1.0 볼륨 그룹(기본값)
-s pe_size	MB 단위의 물리 확장 영역 크기(기본값: 4)
-e max_pe	물리 볼륨당 최대 물리 확장 영역 수(기본값: 1016)
-l max_lv	최대 논리 볼륨 수(기본값: 255)
-p max_pv	최대 물리 볼륨 수(기본값: 255)

물리 볼륨의 크기는 $pe_size \times max_pe$ 로 제한됩니다. 이 볼륨 그룹에 약 4GB($1016 * 4MB$)보다 큰 디스크를 할당하려면 더 큰 pe_size 또는 max_pe 값을 사용하십시오.

각 디스크에서 있는 LVM 메타데이터의 크기는 max_lv , max_pv 및 max_pe 에 따라 달라집니다. `vgcreate` 옵션을 사용할 경우 메타데이터가 사용 가능한 공간을 초과하게 되면 `vgcreate`에서 볼륨 그룹을 만들지 않습니다. 따라서 max_lv , max_pv 및 max_pe 의 새 값을 선택해야 합니다. 예를 들어 100GB보다 큰 디스크를 사용하려면 max_pv 를 줄여야 합니다. 최적 확장 영역 크기 선택에 대한 권장 사항은 **부록 C (161 페이지)**를 참조하십시오.

버전 2.x 볼륨 그룹 만들기

버전 2.x 볼륨 그룹의 경우 `vgcreate` 명령에 물리 볼륨 수(`-p`), 논리 볼륨 수(`-l`) 또는 물리 볼륨당 확장 영역(`-e`)의 최대값이 필요하지 않습니다. 대신 확장 영역 크기(`-s`)와 볼륨 그룹이 증가할 수 있는 최대 크기(`-S`)만 지정해야 합니다. 예를 들면 다음과 같습니다.

```
# vgcreate -V 2.0 -s pe_size -S vg_size /dev/vgname /dev/disk/disk3
```

블록 장치 파일을 사용하여 각 디스크를 볼륨 그룹에 포함합니다. 하나의 명령으로 볼륨 그룹에 모든 물리 볼륨을 할당하거나 단일 물리 볼륨을 사용하여 볼륨 그룹을 만들 수 있습니다. 물리 볼륨은 기존 볼륨 그룹의 일부일 수 없습니다.

다음 옵션을 사용해야 합니다.

-V 2.0 또는 -V 2.1	버전 2.0 또는 버전 2.1 볼륨 그룹
-s pe_size	MB 단위의 물리 확장 영역 크기
-S vg_size	볼륨 그룹의 향후 최대 크기

볼륨 그룹의 크기는 볼륨 그룹에 할당된 모든 물리 볼륨에 있는 사용자 데이터 공간의 합입니다. vg_size 는 만들 때의 볼륨 그룹 크기가 아니며 볼륨 그룹이 향후에 증가할 수 있는 크기입니다. 이 값은 각각 m , g , t 또는 p 문자를 추가하여 메가바이트, 기가바이트, 테라바이트 또는 페타바이트로 지정할 수 있습니다. 예를 들어 최대 크기를 2TB로 지정하려면 `-S 2t`를 사용합니다.

버전 2.x 볼륨 그룹에서 볼륨 그룹의 물리 확장 영역 수에는 아키텍처상의 제한이 있으므로 물리 확장 영역 크기를 선택하면 볼륨 그룹의 최대 크기에 영향을 줍니다. 지정된 물리 확장 영역 크기의 최대 볼륨 그룹 크기를 표시하려면 `-s` 옵션과 함께 `-E` 옵션을 `vgcreate`에 사용합니다. 예를 들면 다음과 같습니다.

```
# vgcreate -V 2.0 -E -s 256
Max_VG_size=2p:extent_size=256m
```

반대로 지정된 볼륨 그룹 크기의 최소 물리 확장 영역 크기를 표시하려면 -s와 함께 -E 옵션을 vgcreate에 사용합니다. 예를 들면 다음과 같습니다.

```
# vgcreate -V 2.0 -E -S 2t
Max_VG_size=2t:extent_size=1m
```

볼륨 그룹에 디스크 추가

대개 새 디스크를 시스템에 추가할 경우 전체 새 볼륨 그룹을 만드는 대신 기존 볼륨 그룹에 추가해야 합니다. 파일 시스템이나 데이터베이스와 같은 사용자 데이터에 대해 새 디스크를 추가하려는 경우 루트 볼륨 그룹에 추가하지 마십시오. 대신 루트 볼륨 그룹은 루트 파일 시스템 및 시스템 파일 시스템(예: /usr, /tmp 등)을 포함하는 디스크로만 유지합니다.

볼륨 그룹에 디스크를 추가하려면 다음 단계를 수행합니다.

1. “LVM 사용을 위한 디스크 초기화” (59 페이지)에 설명된 대로 pvcreate 명령을 사용하여 디스크를 물리 볼륨으로 초기화합니다.
2. vgextend 명령과 디스크의 블록 장치 파일을 사용하여 볼륨 그룹에 물리 볼륨을 추가합니다. 예를 들면 다음과 같습니다.

```
# vgextend /dev/vgname /dev/disk/disk3
```

볼륨 그룹에서 디스크 제거

볼륨 그룹에서 디스크를 제거하려면 다음 단계를 수행합니다.

1. pvdisplay 명령을 사용하여 디스크에 할당된 물리 확장 영역이 없는지 확인합니다. 예를 들면 다음과 같습니다.

```
# pvdisplay /dev/disk/disk3
-- 물리 볼륨 --
PV 이름                /dev/disk/disk3
VG 이름                /dev/vg00
PV 상태              사용 가능
할당 가능              예
VGDA                  2
현재 LV              9
PE 크기(MB)          4
총 PE                1023
사용 가능 PE          494
할당된 PE            529
오래된 PE            0
IO 시간 제한(초)      기본값
자동 전환             켜짐
능동적 폴링           켜짐

-- 물리 볼륨 백업 --
LV 이름                LV의 LE    LV에 대한 PE
/dev/vg00/lvol1        25          25
```

```
/dev/vg00/lvol2    25          25
/dev/vg00/lvol3    50          50
```

--- 물리 확장 영역 ---

PE	상태	LV	LE
0000	현재	/dev/vg00/lvol1	0000
0001	현재	/dev/vg00/lvol1	0001
0002	현재	/dev/vg00/lvol1	0002
1021	사용 가능		0000
1022	사용 가능		0000

사용 가능한 물리 확장 영역 수(사용 가능한 PE)가 총 물리 확장 영역 수(총 PE)와 일치하는지 확인합니다. 일치하지 않을 경우 다음 작업 중 하나를 수행합니다.

- 물리 확장 영역을 볼륨 그룹의 다른 물리 볼륨으로 이동합니다. 자세한 내용은 “다른 물리 볼륨으로 데이터 이동” (84 페이지)을 참조하십시오.
- “논리 볼륨 제거” (69 페이지)의 설명에 따라 디스크에서 논리 볼륨을 제거합니다. 디스크에 있는 논리 볼륨과 물리 확장 영역은 `pvdisplay` 목록 끝에 표시됩니다.

2. 디스크에 물리 확장 영역이 더 이상 포함되지 않으면 `vgreduce` 명령을 사용하여 볼륨 그룹에서 제거합니다. 예를 들면 다음과 같습니다.

```
# vgreduce /dev/vgmn /dev/disk/disk3
```



중요: LVM `pvlinks`를 사용할 경우 “다중 경로 지정을 통해 하드웨어 경로 중복 늘리기” (43 페이지)의 설명에 따라 디스크의 각 링크에 대해 `vgreduce` 명령을 실행해야 합니다.

논리 볼륨 만들기

논리 볼륨을 만들려면 다음 단계를 수행합니다.

1. 논리 볼륨에 필요한 디스크 공간을 결정합니다.

예를 들어 200MB의 장치 스왑 공간을 추가하거나 10GB까지 증가할 것으로 예상되는 새 프로젝트를 추가할 수 있습니다.

2. 사용 가능한 공간이 충분한 볼륨 그룹을 찾습니다.

볼륨 그룹에 논리 볼륨으로 사용할 수 있는 디스크 공간이 충분한지 여부를 확인하려면 `vgdisplay` 명령을 사용하여 이 정보를 계산합니다. `vgdisplay`는 물리 확장 영역 크기(PE 크기(MB)) 및 사용 가능한 물리 확장 영역의 수(사용 가능한 PE)를 포함하여 하나 이상의 볼륨 그룹에 있는 데이터를 출력합니다. 이 두 숫자를 곱하면 볼륨 그룹에서 사용할 수 있는 MB 수를 얻게 됩니다. 자세한 내용은 `vgdisplay(1M)`를 참조하십시오.

3. `lvcreate`를 사용하여 논리 볼륨을 만듭니다. 예를 들면 다음과 같습니다.

```
# lvcreate -L size_in_MB /dev/vgmn
```

이 명령을 사용하면 논리 볼륨 /dev/vgmn/lvoln이 만들어집니다. 이때 LVM이 자동으로 lvoln에 n을 지정합니다.

LVM은 논리 볼륨을 만들 때 해당 논리 볼륨에 대한 블록 및 특성 장치 파일을 만들어 /dev/vgmn 디렉토리에 저장합니다.

스트라이프 논리 볼륨 만들기

스트라이프 논리 볼륨을 만들려면 -i 및 -I 옵션과 함께 lvcreate를 사용하여 각각 디스크 수와 스트라이프 너비를 지정합니다. 예를 들어 디스크 세 개에 걸쳐 크기가 32K인 스트라이프를 사용하려고 합니다. 논리 볼륨 크기는 240MB입니다. 스트라이프 논리 볼륨을 만들려면 다음 명령을 입력합니다.

```
# lvcreate -i 3 -I 32 -l 240 -n lvol11 /dev/vg01
```

lvcreate 명령은 자동으로 디스크 개수와 확장 영역 크기를 곱한 값의 배수로 논리 볼륨의 크기를 올립니다. 예를 들어 디스크 세 개에 걸쳐 스트라이프를 사용하고 확장 영역 크기가 4MB인 경우 논리 볼륨의 크기를 200MB(-l 200)로 지정해도 200은 12의 배수가 아니므로 lvcreate는 204MB의 논리 볼륨을 만듭니다.



참고: 여러 디스크에 걸쳐 스트라이프를 사용할 때 스트라이프 볼륨 크기는 가장 작은 디스크의 용량과 스트라이핑에 사용된 디스크 개수를 곱한 수를 초과할 수 없습니다.

미러 논리 볼륨 만들기

미러 논리 볼륨을 만들려면 -m 옵션과 함께 lvcreate를 사용하여 미러 사본의 수를 선택합니다. 미러 사본이 관리되는 방식을 제어하려면 다음 옵션 중에서 선택합니다.

엄격, 비엄격 또는 PVG 엄격 확장 영역 할당

-s y 엄격한 할당(기본값)
-s n 비엄격 할당
-s g PVG 엄격 할당

연속 또는 비연속 확장 영역 할당

-C y 연속 할당
-C n 비연속 할당(기본값)

미러 예약 정책

-d p 병렬 예약(기본값)
-d s 순차 예약

미러 일관성 정책

-M y MWC 사용(기본값, 크래시 복구 중 최적 미러 재동기화)
-M n -c y MCR 사용(크래시 복구 중 전체 미러 재동기화)
-M n -c n MCR 사용 안 함(크래시 복구 중 미러 재동기화 안 함)

예를 들어 하나의 미러 사본, 비엄격 할당, 병렬 예약 및 미러 재동기화 안 함으로 240MB의 미러 논리 볼륨을 만들려면 다음 명령을 입력합니다.

```
# lvcreate -m 1 -s n -d p -M n -c n -L 240 -n lvol11 /dev/vg01
```




팁: 기존 미러 논리 볼륨의 특성을 변경하려면 `lvchange` 명령을 사용합니다. 이 명령은 `-C`, `-c`, `-d`, `-M` 및 `-s` 옵션을 지원합니다. 자세한 내용은 `lvchange(1M)`을 참조하십시오.

논리 볼륨 확장



참고: 논리 볼륨에 공간을 추가해도 논리 볼륨을 사용하는 엔터티에 해당 공간이 자동으로 할당되지 않습니다. 예를 들어, 논리 볼륨에 포함된 파일 시스템에 공간을 추가하려면 논리 볼륨을 확장한 후 `extendfs`를 실행해야 합니다. 자세한 내용은 “파일 시스템 논리 볼륨 관리” (105 페이지) 및 “스왑 논리 볼륨 관리” (111 페이지)를 참조하십시오.

1. 논리 볼륨에 더 필요한 추가 디스크 공간을 결정합니다.
예를 들어 200MB의 스왑 공간을 추가하거나 기존 프로젝트에 추가로 1GB가 필요할 수 있습니다.
2. `vgdisplay` 명령을 사용하여 사용 가능한 공간을 찾습니다. 예를 들면 다음과 같습니다.

```
# vgdisplay vg00
--- 볼륨 그룹 ---
VG 이름                /dev/vg00
VG 쓰기 권한          읽기/쓰기
VG 상태              사용 가능
최대 LV                255
현재 LV                8
열린 LV               8
최대 PV               16
열린 PV                1
활성 PV                1
PV당 최대 PE          2000
VGDA                  2
PE 크기(MB)            4
총 PE                 249
할당된 PE             170
사용 가능한 PE        79
총 PVG                 0
총 스페어 PV           0
사용 중인 총 스페어 PV  0
VG 버전               1.0
VG 최대 크기           1082g
VG 최대 확장 영역       69248
```

사용 가능한 PE 항목은 4MB 크기의 사용 가능한 확장 영역 개수를 나타냅니다. 이 경우에는 79개이며 316MB와 같습니다.

3. 논리 볼륨을 확장합니다. 예를 들면 다음과 같습니다.

```
# lvextend -L 332 /dev/vg00/lvol1
```

명령은 이 볼륨의 크기를 332MB로 증가시킵니다.

논리 볼륨을 특정 디스크로 확장

성능상의 이유로 논리 볼륨을 여러 디스크에 분배할 수 있습니다. 예를 들어 30GB의 논리 볼륨을 만들어 첫 번째 디스크에 10GB를 넣고 두 번째 디스크에 다른 10GB를, 세 번째 디스크에 10GB를 넣으려는 경우 확장 영역 크기가 4MB라고 가정하면 논리 볼륨에는 총 7680개의 확장 영역이 필요합니다. 논리 볼륨을 확장하려면 다음 단계를 수행합니다.

1. 디스크에 물리 볼륨을 만들고 볼륨 그룹을 만든 다음 `lv011`이라는 크기가 0인 논리 볼륨을 만듭니다. 예를 들면 다음과 같습니다.

```
# lvcreate -n lv011 /dev/vg01
```
2. 확장 영역의 1/3을 첫 번째 물리 볼륨의 논리 볼륨에 할당합니다. 예를 들면 다음과 같습니다.

```
# lvextend -l 2560 /dev/vg01/lv011 /dev/disk/disk7
```
3. 나머지 물리 볼륨에 대한 논리 볼륨에 할당된 물리 확장 영역의 총 수를 2560으로 늘립니다. 각각의 경우 추가적으로 2560개의 확장 영역이 지정된 디스크에 할당됩니다. 예를 들면 다음과 같습니다.

```
# lvextend -l 5120 /dev/vg01/lv011 /dev/disk/disk8
# lvextend -l 7680 /dev/vg01/lv011 /dev/disk/disk9
```

`lvextend`에 `-l` 옵션을 사용할 때 논리 확장 영역에 공간을 지정합니다.

다른 예로 볼륨 그룹에 모델이 동일한 디스크가 두 개 있다고 가정합니다. 현재 한 디스크에만 상주하는 논리 볼륨 24GB가 있습니다. 논리 볼륨의 크기를 40GB로 확장하여 다른 디스크에 16GB를 추가적으로 할당하려고 합니다.

다음과 같이 논리 볼륨을 특정 디스크로 확장합니다.

```
# lvextend -L 40960 /dev/vg01/lv012 /dev/disk/disk3
```

여기서 `-L` 옵션(대문자)을 사용하는 경우 논리 확장 영역 수가 아닌 MB로 공간을 지정합니다.

명령 옵션에 대한 자세한 내용은 `lvextend(1M)`를 참조하십시오.

논리 볼륨 축소



주의: 논리 볼륨을 축소하기 전에 해당 논리 볼륨의 사용자에게 알려야 합니다.

예를 들어, 파일 시스템이 들어 있는 논리 볼륨을 축소시키기 전에 **파일 시스템을 백업하십시오**. 현재 파일 시스템이 논리 볼륨의 새로운(축소된) 크기보다 적은 공간을 차지하고 있더라도 논리 볼륨을 축소시키면 데이터가 손실됩니다. 파일 시스템 및 스왑 장치에 적절한 절차를 보려면 “파일 시스템 논리 볼륨 관리” (105 페이지) 및 “스왑 논리 볼륨 관리” (111 페이지)를 참조하십시오.

논리 볼륨을 축소하려면 다음 단계를 수행합니다.

1. 논리 볼륨을 사용 중인 응용 프로그램을 찾으려면 `fuser` 명령을 사용합니다. 예를 들면 다음과 같습니다.

```
# fuser -cu /dev/vg01/lvol5
```

논리 볼륨이 사용 중이면 기본 응용 프로그램에서 크기 축소를 처리할 수 있는지 확인합니다. 응용 프로그램을 중지해야 할 수 있습니다.
2. 논리 볼륨의 새 크기를 결정합니다.
예를 들어 논리 볼륨이 파일 시스템에 마운트된 경우 새로운 크기는 파일 시스템의 데이터가 현재 차지하고 있는 공간보다 커야 합니다. `bdf` 명령을 실행하면 마운트된 모든 볼륨의 크기가 표시됩니다. 첫 번째 열에는 볼륨에 할당된 공간이 표시되고 두 번째 열에는 실제 사용되고 있는 공간이 표시됩니다. 논리 볼륨의 새로운 크기는 `bdf` 출력의 두 번째 열에 표시된 크기보다 더 커야 합니다.
3. 다음과 같이 논리 볼륨의 크기를 축소합니다.

```
# lvreduce -L 500 /dev/vg01/lvol5
```

이 명령은 논리 볼륨 `/dev/vg01/lvol5`를 500MB로 축소시킵니다.

논리 볼륨에 미리 추가



참고: 미러링에는 선택적 제품인 HP MirrorDisk/UX가 필요합니다.



팁: HP SMH를 사용하여 이 작업을 더 쉽게 수행할 수 있습니다. HP SMH에서는 미러 사본에 사용 가능한 디스크 공간이 충분한지 그리고 사용 가능한 공간이 할당 정책에 맞는지 확인합니다.

논리 볼륨에 미러를 추가하려면 다음 단계를 수행합니다.

1. 만들려는 미러 사본의 개수를 결정합니다.
이 예제에서는 미러를 하나 만듭니다. 즉, 온라인 데이터가 두 개(원본과 미러 사본 한 개)가 됩니다.
2. 미러링할 논리 볼륨이 들어 있는 볼륨 그룹에 사용 가능한 공간이 충분한지 확인합니다.
볼륨 그룹에는 미러링할 논리 볼륨만큼 여분의 공간이 현재 할당되어 있어야 합니다. 즉, 이 볼륨에 필요한 실제 공간의 두 배가 되어야 합니다.
엄격한 미러링(별도의 디스크에 미러 사본을 보유하므로 HP에서 권장함)을 사용하려면 현재 미러링할 볼륨에서 사용하지 않는 디스크에 이 여분의 공간이 있어야 합니다.
3. `lvextend` 명령에 `-m` 옵션을 사용하여 원하는 추가 사본 수를 추가합니다. 예를 들면 다음과 같습니다.

```
# lvextend -m 1 /dev/vg00/lvol1
```

이 명령은 지정된 논리 볼륨의 단일 미러 사본을 추가합니다.

특정 물리 볼륨에 미러 사본을 적용하려면 명령줄 끝에 미러 사본을 추가합니다. 예를 들면 다음과 같습니다.

```
# lvextend -m 1 /dev/vg00/lvol1 /dev/disk/disk4
```

논리 볼륨에서 미러 제거

미러 사본을 제거하려면 `lvreduce` 명령을 사용하여 남겨 둘 미러 사본 수를 지정합니다. 예를 들어 논리 볼륨의 모든 미러를 제거하려면 다음 명령을 입력합니다.

```
# lvreduce -m 0 /dev/vg00/lvol1
```

이렇게 하면 미러 사본 수가 0으로 축소되므로 원본만 남게 됩니다.

특정 디스크에서 미러 사본을 제거하려면 `lvreduce`를 사용하고 미러 사본을 제거할 디스크를 지정합니다. 예를 들면 다음과 같습니다.

```
# lvreduce -m 0 /dev/vg00/lvol1 /dev/disk/disk4
```

논리 볼륨 이름 바꾸기

논리 볼륨의 이름을 변경하려면 다음 단계를 수행합니다.

1. 논리 볼륨에 두 가지 기존 장치 파일 즉, 블록 장치 파일과 특성 또는 원시 장치 파일이 있는지 확인합니다. 특성 장치 파일의 이름이 `r`로 시작한다는 점을 제외하면 이러한 장치 파일의 이름은 동일합니다. 예를 들어 볼륨 그룹 `vg00`에 속한 한 논리 볼륨의 이름을 `lvol1`에서 `database`로 바꾸려면 `/dev/vg00` 디렉토리의 내용을 나열합니다. 예를 들면 다음과 같습니다.

```
# cd /dev/vg00
# ls -l
total 0
crw-r----- 1 root      sys   64 0x0000000 Nov 16 02:49 group
brw-r----- 1 root      sys   64 0x0000001 Nov 16 02:49 lvol1
brw-r----- 1 root      sys   64 0x0000002 Nov 16 02:49 lvol2
brw-r----- 1 root      sys   64 0x0000003 Nov 16 02:49 lvol3
brw-r----- 1 root      sys   64 0x0000004 Nov 16 02:49 lvol4
crw-r----- 1 root      sys   64 0x0000001 Nov 16 02:49 rlvol1
crw-r----- 1 root      sys   64 0x0000002 Nov 16 02:49 rlvol2
crw-r----- 1 root      sys   64 0x0000003 Nov 16 02:49 rlvol3
crw-r----- 1 root      sys   64 0x0000004 Nov 16 02:49 rlvol4
```

2. 두 파일의 이름을 모두 바꾸려면 `mv` 명령을 사용합니다. 예를 들면 다음과 같습니다.

```
# mv /dev/vg00/lvol1 /dev/vg00/database
# mv /dev/vg00/rlvol1 /dev/vg00/rdatabase
```

3. 시스템에 있는 다른 모든 파일에서 기존 이름에 대한 참조를 모두 업데이트합니다. 여기에는 마운트된 파일 시스템이나 스왑 장치에 대한 `/etc/fstab`와 `vgexport` 명령의 기존 맵 파일이 포함됩니다.

논리 볼륨 제거



주의: 논리 볼륨을 제거하면 해당 내용을 사용할 수 없고 덮어쓰여지게 됩니다. 특히 논리 볼륨에 들어 있는 파일 시스템이 손상됩니다.

논리 볼륨을 제거하려면 다음 단계를 수행합니다.

1. 논리 볼륨을 파일 시스템 또는 응용 프로그램의 원시 디스크 공간으로 사용 중이 아닌지 확인합니다. 다음과 같이 `fuser` 명령을 사용합니다.

```
# fuser -cu /dev/vg01/lvol5
```

논리 볼륨이 사용 중이면 기본 응용 프로그램에 해당 논리 볼륨이 더 이상 필요하지 않은지 확인합니다. 응용 프로그램을 중지해야 할 수 있습니다.

2. `lvremove` 명령을 사용하여 논리 볼륨을 제거합니다. 예를 들면 다음과 같습니다.

```
# lvremove /dev/vg01/lvol5
```

이제 이 공간을 사용하여 기존 논리 볼륨을 확장하거나 새 논리 볼륨을 만들 수 있습니다.

볼륨 그룹 내보내기

볼륨 그룹을 내보내면 볼륨 그룹에 대한 데이터가 시스템에서 모두 제거되지만 디스크에 있는 데이터는 그대로 유지됩니다. 내보낸 볼륨의 디스크를 실제로 이동하거나 다른 시스템에 연결할 수 있고 볼륨 그룹을 해당 위치로 가져올 수 있습니다.

볼륨 그룹을 내보내면 볼륨 그룹 및 관련 물리 그룹에 대한 정보가 `/etc/lvmtab` 및 `/etc/lvmtab.p`에서 제거되고 장치 파일이 있는 볼륨 그룹의 디렉토리가 `/dev` 디렉토리에서 제거됩니다.

1. 볼륨 그룹의 논리 볼륨이 사용 중이 아닌지 확인합니다. 볼륨 그룹의 논리 볼륨을 사용하는 응용 프로그램을 중지하고 볼륨 그룹에 들어 있는 파일 시스템을 마운트 해제해야 할 수 있습니다.

각 논리 볼륨에 대해 `fuser` 명령을 사용합니다. 예를 들면 다음과 같습니다.

```
# fuser -cu /dev/vgnn/lvoln
```

2. 볼륨 그룹을 비활성화합니다. 예를 들면 다음과 같습니다.

```
# vgchange -a n vgnn
```

3. `vgexport` 명령을 사용하여 볼륨 그룹을 내보냅니다. 예를 들면 다음과 같습니다.

```
# vgexport -v -m /tmp/vgnn.map vgnn
```

볼륨 그룹을 다른 시스템으로 이동하려면 `vgexport`에 `-m` 옵션을 사용하여 맵 파일을 만듭니다. 이 ASCII 파일은 디스크에 저장되지 않으므로 논리 볼륨 이름을 포함합니다. 볼륨 그룹의 논리 볼륨에 기본 이름 `/dev/vgnn/lvoln`을 사용하지 않으려면 맵 파일을 만들어야 합니다.

볼륨 그룹에 디스크가 여러 개 있으면 `vgexport`와 `-s` 옵션을 함께 사용합니다. 이 옵션은 VGID(볼륨 그룹 식별자)를 맵 파일에 추가합니다. 볼륨 그룹을 가져올 때

모든 디스크를 이름별로 지정하지 않을 수도 있습니다. 자세한 내용은 “볼륨 그룹 가져오기” (70 페이지)를 참조하십시오.

vgexport가 완료되면 볼륨 그룹에 대한 모든 정보가 시스템에서 제거됩니다. 이제 디스크를 다른 시스템으로 이동할 수 있고 볼륨 그룹을 해당 위치로 가져올 수 있습니다.

볼륨 그룹 가져오기

볼륨 그룹을 가져오려면 다음 단계를 수행합니다.

1. 시스템에 디스크를 연결합니다.
2. 2008년 3월 이전의 HP-UX 릴리즈를 사용하고 있거나 볼륨 그룹 장치 파일의 보조 번호를 지정하려면 “볼륨 그룹 장치 파일 만들기” (60 페이지)의 절차를 사용하여 만듭니다.
3. vgimport 명령을 사용하여 볼륨 그룹을 가져옵니다.

```
# vgimport -v -N -m /tmp/vgmn.map /dev/vgmn list_of_disks
```

볼륨 그룹에 디스크가 여러 개 있고 VGID가 맵 파일에 저장된 경우(즉, vgexport 명령이 -s 및 -m 옵션과 함께 실행된 경우) -s 옵션을 사용하여 vgimport 명령줄에서 해당 디스크를 모두 지정하지 않아도 됩니다. 이렇게 하면 vgimport에서 시스템에 있는 모든 디스크를 스캔합니다. VGID가 맵 파일의 VGID와 일치하는 물리 볼륨은 볼륨 그룹에 자동으로 포함됩니다.

4. 다음과 같이 볼륨 그룹을 활성화합니다.

```
# vgchange -a y vgmn
```



참고: 볼륨 그룹에 다중 경로 지정된 디스크가 있으면 LVM 대체 링크의 수퍼셋인 HP-UX의 기본 다중 경로 지정을 사용하는 것이 좋습니다. 자세한 내용은 “다중 경로 지정을 통해 하드웨어 경로 중복 늘리기” (43 페이지)를 참조하십시오.

LVM의 대체 링크 기능을 사용하려면 볼륨 그룹을 가져올 때 몇 가지 관련 사항이 있습니다.

- vgimport 명령에 대해 -N 옵션을 생략해야 합니다.
- vgimport는 발견된 첫 번째 링크를 모든 물리 볼륨의 기본 링크로 설정합니다. 가져오기 후에 링크가 원하는 순서가 아닌 경우 기본을 변경할 각 물리 볼륨의 기본 링크에 대해 vgreduce와 vgextend를 사용합니다.
- 튜너블 maxfiles는 사용 가능한 디스크 수의 두 배가 넘어야 합니다.

볼륨 그룹 매개 변수 수정



참고: vgmodify 명령은 버전 2.x 볼륨 그룹을 지원하지 않습니다.

볼륨 그룹을 만들 때 물리 볼륨당 최대 물리 확장 영역 수, 최대 물리 볼륨 수 및 최대 논리 볼륨 수와 같은 볼륨 그룹의 특정 특성을 설정합니다. vgmodify 명령을 사용하면 볼륨 그룹을 제거한 후 다시 만들거나 데이터를 이동하지 않고서도 이러한 매개 변수를 조정할 수 있습니다.

다음 절차를 수행하여 이러한 볼륨 그룹 매개 변수를 조정합니다.

1. `vgmodify`를 실행하여 볼륨 그룹에 대한 정보를 수집합니다.

다음과 같은 세 가지 명령의 출력을 저장합니다.

```
# vgmodify -o -r vgnn
# vgmodify -v -t vgnn
# vgmodify -v -n -t vgnn
```

-o 옵션은 기존 LVM 메타데이터 공간을 활용하여 값을 최적화합니다. -t 옵션은 물리 확장 영역의 번호를 다시 매기지 않고 최적화된 설정 영역을 보고합니다. -n 옵션은 물리 확장 영역의 번호 다시 매기기 기능을 활성화합니다.

2. 이전 단계에서 수집된 정보에 따라 볼륨 매개 변수의 새 값을 선택합니다.
3. 새 값에 따라 각 물리 볼륨의 VGRA(Volume Group Reserved Area) 크기가 증가할 수 있습니다. VGRA는 LVM 헤더에 상주하므로 크기를 늘리려면 물리 볼륨에 있는 사용자 데이터의 첫 번째 물리 확장 영역을 이동해야 합니다. `pvmove` 명령을 사용하여 첫 번째 물리 확장 영역을 다른 위치로 이동합니다.
4. `vgmodify`를 새 설정 및 -r 옵션과 함께 실행하여 값을 검토합니다.
5. 볼륨 그룹을 비활성화합니다.
6. -r 옵션 없이 `vgmodify`를 실행하여 새 값을 커밋합니다.
7. 볼륨 그룹을 활성화합니다. `vgdisplay` 명령을 실행하여 설정이 적용되었는지 확인합니다.

예를 들어, 보다 큰 디스크를 볼륨 그룹 `vg32`에 추가하려고 합니다. 물리 볼륨당 최대 물리 확장 영역 수(`max_pe`) 및 최대 물리 볼륨 수(`max_pv`)를 늘리려고 합니다. 다음과 같은 단계를 수행합니다.

1. `vgmodify`를 실행하여 볼륨 그룹에 대한 정보를 수집합니다.

다음과 같은 세 가지 명령의 출력을 저장합니다.

```
# vgmodify -o -r vg32
현재 볼륨 그룹 설정:
```

최대 LV	255
최대 PV	16
PV당 최대 PE	1016
PE 크기 (MB)	32
VGRA 크기 (KB)	176

새 구성에서는 "max_pes"이(가) 1016에서 6652으로 증가되었습니다.
현재 값과 새 볼륨 그룹 매개 변수가 다릅니다.
볼륨 그룹에 대한 업데이트가 필요합니다.

새 볼륨 그룹 설정:

최대 LV	255
최대 PV	16
PV당 최대 PE	6652
PE 크기 (MB)	32
VGRA 크기 (KB)	896

검토를 완료했습니다. 볼륨 그룹이 수정되지 않았습니다.

```
# vgmodify -v -t vg32
현재 볼륨 그룹 설정:
```

최대 LV	255
-------	-----

최대 PV	16
PV당 최대 PE	1016
PE 크기 (MB)	32
VGRA 크기 (KB)	176

확장 영역을 사용 중인 물리 볼륨의 VGRA 공간 (KB) :

PV	현재	-n
/dev/rdisk/disk6	896	32768
/dev/rdisk/disk5	896	32768
요약	896	32768

볼륨 그룹 최적화된 설정 (번호 재지정된 PE 없음) :

max_pv(-p)	max_pe(-e)	디스크 크기 (Mb)
2	53756	1720193
3	35836	1146753
...		
213	296	9473
255	252	8065

```
# vgmodify -v -n -t vg32
/dev/vg32에 대한 볼륨 그룹 구성이
/etc/lvmconf/vg32.conf에 저장되었습니다.
```

현재 볼륨 그룹 설정:

최대 LV	255
최대 PV	16
PV당 최대 PE	1016
PE 크기 (MB)	32
VGRA 크기 (KB)	176

확장 영역을 사용 중인 물리 볼륨의 VGRA 공간 (KB) :

PV	현재	-n
/dev/rdisk/disk6	896	32768
/dev/rdisk/disk5	896	32768
요약	896	32768

물리 확장 영역 0이 모든 PV에서 사용 가능하지는 않습니다. 다음의 모든 디스크에서 첫 번째 확장 영역이 사용 가능해야 (pvmove (1M) 참조) 이러한 값을 얻을 수 있습니다.

```
/dev/rdisk/disk6
/dev/rdisk/disk5
```

볼륨 그룹 최적화된 설정 (PE가 낮게 번호 재지정됨) :

max_pv(-p)	max_pe(-e)	디스크 크기 (Mb)
61	65535	2097152
62	65532	2097056
...		
252	16048	513568
255	15868	507808

2. vgmodify -n -t의 출력에 따라 max_pv에 대해 255를, max_pe에 대해 15868을 선택합니다.
3. 새 값은 물리 확장 영역 0을 사용 가능하게 해야 하므로 pvmove를 사용하여 물리 확장 영역을 다른 위치로 이동합니다.

```
# pvmove /dev/disk/disk5:0 /dev/disk/disk5
논리 볼륨 "/dev/vg32/lvol2"의 논리 확장 영역을 전송하는 중...
물리 볼륨 "/dev/disk/disk5"을 (를) 성공적으로 이동했습니다.
```


/dev/vg32에 대한 볼륨 그룹 구성이
/etc/lvmconf/vg32.conf에 저장되었습니다.

```
# pvmove /dev/disk/disk6:0 /dev/disk/disk6
```

논리 볼륨 "/dev/vg32/lvol1"의 논리 확장 영역을 전송하는 중...

물리 볼륨 "/dev/disk/disk6"을(를) 성공적으로 이동했습니다.

/dev/vg32에 대한 볼륨 그룹 구성이
/etc/lvmconf/vg32.conf에 저장되었습니다.

4. vgmodify에 -r 옵션을 사용하여 변경 내용을 미리 봅니다.

```
# vgmodify -p 255 -e 15868 -r -n vg32
```

현재 볼륨 그룹 설정:

최대 LV	255
최대 PV	16
PV당 최대 PE	1016
PE 크기(MB)	32
VGRA 크기(KB)	176

현재 값과 새 볼륨 그룹 매개 변수가 다릅니다.
볼륨 그룹에 대한 업데이트가 필요합니다.

새 볼륨 그룹 설정:

최대 LV	255
최대 PV	255
PV당 최대 PE	15868
PE 크기(MB)	32
VGRA 크기(KB)	32640

검토를 완료했습니다. 볼륨 그룹이 수정되지 않았습니다.

5. 볼륨 그룹을 비활성화합니다.

```
# vgchange -a n vg32
```

볼륨 그룹 "vg32"을(를) 성공적으로 변경했습니다.

6. 새 값을 커밋합니다.

```
# vgmodify -p 255 -e 15868 -n vg32
```

현재 볼륨 그룹 설정:

최대 LV	255
최대 PV	16
PV당 최대 PE	1016
PE 크기(MB)	32
VGRA 크기(KB)	176

현재 값과 새 볼륨 그룹 매개 변수가 다릅니다.
볼륨 그룹에 대한 업데이트가 필요합니다.

새 볼륨 그룹 설정:

최대 LV	255
최대 PV	255
PV당 최대 PE	15868
PE 크기(MB)	32
VGRA 크기(KB)	32640

"vg32"에 대한 새 볼륨 그룹 구성이
"/etc/lvmconf/vg32.conf"에 저장되었습니다.
"vg32"에 대한 이전 볼륨 그룹 구성이
"/etc/lvmconf/vg32.conf.old"에 저장되었습니다.
모든 물리 볼륨에 써서 수정을 시작하는 중입니다.
"/etc/lvmconf/vg32.conf"에서 모든 물리 볼륨으로
설정을 적용 중입니다.
수정 프로세스를 완료했습니다.
"vg32"에 대한 새 볼륨 그룹 구성이
"/etc/lvmconf/vg32.conf.old"에 저장되었습니다.
볼륨 그룹 "vg32"을(를) 성공적으로 변경했습니다.

7. 볼륨 그룹을 활성화하고 변경 내용을 확인합니다.

```
# vgchange -a y vg32
```

활성화된 볼륨 그룹

볼륨 그룹 "vg32"을(를) 성공적으로 변경했습니다.

```
# vgdisplay vg32
--- 볼륨 그룹 ---
VG 이름                               /dev/vg32
VG 쓰기 권한                         읽기/쓰기
VG 상태                             사용 가능
최대 LV                             255
현재 LV                              0
열린 LV                              0
최대 PV                             255
열린 PV                              2
활성 PV                              2
PV당 최대 PE                        15868
VGDA                                 4
PE 크기(MB)                         32
총 PE                               1084
할당된 PE                           0
사용 가능한 PE                     1084
총 PVG                              0
총 스페어 PV                        0
사용 중인 총 스페어 PV              0
VG 버전                             1.0
```

볼륨 그룹 정지 및 다시 시작

디스크 관리 유틸리티를 사용하여 볼륨 그룹에 있는 모든 디스크의 백업 이미지나 "스냅샷"을 만들려면 스냅샷을 가져올 때 LVM이 디스크에 쓰는 중이 아닌지 확인해야 합니다. 그렇지 않으면 일부 디스크에 부분적으로 기록되거나 일치하지 않는 LVM 메타데이터가 포함될 수 있습니다. 볼륨 그룹 디스크 이미지를 일관된 상태로 유지하려면 볼륨 그룹을 비활성화하거나 정지해야 합니다.

볼륨 그룹을 비활성화하려면 볼륨 그룹에 있는 손상될 수 있는 논리 볼륨을 모두 닫아야 합니다. 예를 들어, 볼륨 그룹의 논리 볼륨을 사용하는 파일 시스템을 마운트 해제해야 합니다. 그러나 볼륨 그룹을 일시 정지하면 스냅샷 작업 동안 볼륨 그룹은 활성 상태로, 논리 볼륨은 열린 상태로 유지되어 시스템에 미치는 영향을 최소화할 수 있습니다.

볼륨 그룹에 대한 읽기 및 쓰기 작업을 모두 정지하거나 쓰기 작업만 정지할 수 있습니다. 볼륨 그룹이 정지된 동안 `vgdisplay` 명령은 해당 볼륨 그룹의 액세스 모드를 `quiesced`로 보고합니다. 그러나 볼륨 그룹이 다시 시작할 때까지 표시된 I/O 작업이 정지되고 볼륨 그룹 구성을 수정하는 명령이 즉시 실패합니다.



참고: 개별 물리 볼륨 또는 논리 볼륨은 이 기능을 사용하여 정지할 수 없습니다. 물리 볼륨을 일시 정지하여 비활성화하거나 교체하려면 “물리 볼륨에 대한 경로 비활성화” (94 페이지)를 참조하십시오. 논리 볼륨을 정지하려면 볼륨 그룹을 정지하거나 비활성화합니다. 볼륨 그룹을 비활성화하지 않고 논리 볼륨의 안정적인 이미지를 제공하려면 “미러 논리 볼륨 백업” (79 페이지)의 설명대로 논리 볼륨을 미러링한 다음 미러 중 하나를 분리합니다.

볼륨 그룹 정지는 다시 부팅하면 유지되지 않습니다.

볼륨 그룹을 정지하려면 다음과 같이 -Q 옵션과 함께 vgchange 명령을 사용합니다.

```
# vgchange -Q mode vgnn
```

mode 매개 변수는 읽기 및 쓰기 작업을 모두 차단하는 rw 또는 읽기 작업을 허용하지만 쓰기 작업을 차단하는 w가 될 수 있습니다.

기본적으로 볼륨 그룹은 명시적으로 다시 시작될 때까지 정지되어 있습니다. -t 옵션을 사용하여 최대 정지 시간(초)을 지정할 수 있습니다. 정지 시간이 만료되면 볼륨 그룹이 자동으로 다시 시작됩니다. 예를 들어 읽기 작업을 허용하면서 볼륨 그룹 vg08을 최대 10분(600초) 동안 정지하려면 다음 명령을 입력합니다.

```
# vgchange -Q w -t 600 vg08
```

정지된 볼륨 그룹을 다시 시작하려면 다음과 같이 -R 옵션과 함께 vgchange 명령을 사용합니다.

```
# vgchange -R vgnn
```

볼륨 그룹 이름 바꾸기

볼륨 그룹의 이름을 바꾸려면 이 그룹을 내보낸 다음 새 이름으로 가져옵니다. 볼륨 그룹 내보내기 및 가져오기 방법에 대한 자세한 내용은 “볼륨 그룹 내보내기” (69 페이지) 및 “볼륨 그룹 가져오기” (70 페이지)를 참조하십시오.

볼륨 그룹 이름을 vg01에서 vgdb로 바꾸려면 다음 단계를 수행합니다.

1. 다음과 같이 볼륨 그룹을 비활성화합니다.

```
# vgchange -a n vg01
```

2. 볼륨 그룹에 동일한 보조 번호를 유지하려면 다음과 같이 볼륨 그룹의 group 파일을 검토합니다.

```
# ls -l /dev/vg01/group
```

```
crw-r--r-- 1 root sys 64 0x010000 Mar 28 2004 /dev/vg01/group
```

이 예제에서 볼륨 그룹 주 번호는 64이고 보조 번호는 0x010000입니다.

3. 다음과 같이 볼륨 그룹을 내보냅니다.

```
# vgexport -m vg01.map vg01
```

4. 2008년 3월 이전의 HP-UX 릴리즈를 사용하고 있거나 볼륨 그룹 장치 파일의 보조 번호를 지정하려면 “볼륨 그룹 장치 파일 만들기” (60 페이지)의 절차를 사용하여 볼륨 그룹의 새 이름으로 만듭니다.
이 예제에서 group 파일의 주 번호는 64이고 보조 번호는 0x010000이므로 다음 명령을 입력합니다.

```
# mkdir /dev/vgdb
# mknod /dev/vgdb/group c 64 0x010000
```
5. 다음과 같이 새 이름으로 볼륨 그룹을 가져옵니다.

```
# vgimport -m vg01.map /dev/vgdb
```
6. 다음과 같이 볼륨 그룹 구성 정보를 백업합니다.

```
# vgcfgbackup /dev/vgdb
```
7. 다음과 같이 볼륨 그룹을 활성화합니다.

```
# vgchange -a y /dev/vgdb
```
8. 다음과 같이 기존 볼륨 그룹 이름을 기반으로 하여 저장된 구성 정보를 제거합니다.

```
# rm /etc/lvmconf/vg01.conf
```
9. 시스템에 있는 다른 모든 파일에서 기존 이름에 대한 참조를 모두 업데이트합니다. 여기에는 마운트된 파일 시스템이나 스왑 장치에 대한 /etc/fstab와 vgexport 명령의 기존 맵 파일이 포함됩니다.

볼륨 그룹 분리

분리할 물리 볼륨이 자체 포함되어 있는 경우 즉, 물리 볼륨의 논리 볼륨이 해당 물리 볼륨에 완전히 포함되어 있어야 하는 경우 `vgchgid`를 사용하여 기존 볼륨 그룹을 두 개 이상의 볼륨 그룹으로 분리할 수 있습니다. 예를 들어 분리 가능한 볼륨 그룹의 물리 볼륨 0, 1에 논리 볼륨 1, 2, 3이 있고 물리 볼륨 2, 3, 4, 5에 논리 볼륨 4, 5, 6이 있을 수 있습니다.

이 예제에서 볼륨 그룹 `vgold`에는 물리 볼륨 `/dev/disk/disk0`부터 `/dev/disk/disk5`가 들어 있습니다. 논리 볼륨 `lv011`, `lv012`, `lv013`은 물리 볼륨 `/dev/disk/disk0`과 `/dev/disk/disk1`에 있고 논리 볼륨 `lv014`, `lv015` 및 `lv016`은 나머지 물리 볼륨에 있습니다.

`/dev/disk/disk0` 및 `/dev/disk/disk1`을 `vgold`에 보관하고 나머지 물리 볼륨을 `vgnew`라는 새 볼륨 그룹으로 분리하려면 다음 단계를 수행합니다.

1. 다음과 같이 볼륨 그룹을 비활성화합니다.

```
# vgchange -a n vgold
```
2. 다음과 같이 볼륨 그룹을 내보냅니다.

```
# vgexport vgold
```
3. 다음과 같이 새 볼륨 그룹에 할당할 물리 볼륨의 VGID를 변경합니다.

```
# vgchgid -f /dev/rdisk/disk2 /dev/rdisk/disk3 \
    /dev/rdisk/disk4 /dev/rdisk/disk5
```

4. 2008년 3월 이전의 HP-UX 릴리즈를 사용하고 있거나 vgold 그룹 파일의 보조 번호를 지정하려면 “볼륨 그룹 장치 파일 만들기” (60 페이지)의 절차를 사용하여 만듭니다.
5. 2008년 3월 이전의 HP-UX 릴리즈를 사용하고 있거나 vgnew 그룹 파일의 보조 번호를 지정하려면 “볼륨 그룹 장치 파일 만들기” (60 페이지)의 절차를 사용하여 만듭니다.
6. 다음과 같이 이전 볼륨 그룹의 물리 볼륨을 가져옵니다.

```
# vgimport /dev/vgold /dev/rdisk/disk0 /dev/rdisk/disk1
```
7. 다음과 같이 새 볼륨 그룹의 물리 볼륨을 가져옵니다.

```
# vgimport /dev/vgnew /dev/rdisk/disk2 /dev/rdisk/disk3 \
/dev/rdisk/disk4 /dev/rdisk/disk5
```
8. 볼륨 그룹을 활성화합니다. 다음과 같이 이전 볼륨 그룹의 디스크 중 절반 이상이 누락되므로 해당 그룹에 대한 쿼럼 검사를 비활성화합니다.

```
# vgchange -a y -q n /dev/vgold
# vgchange -a y /dev/vgnew
```
9. 논리 볼륨은 현재 두 볼륨 그룹에 모두 정의되어 있습니다. 다음과 같이 복제 논리 볼륨을 더 이상 포함하지 않는 볼륨 그룹에서 해당 논리 볼륨을 제거합니다.

```
# lvremove -f vgold/lvol4 vgold/lvol5 vgold/lvol6
# lvremove -f vgnew/lvol1 vgnew/lvol2 vgnew/lvol3
```
10. 물리 볼륨은 현재 두 볼륨 그룹에 모두 정의되어 있습니다. 다음과 같이 두 볼륨 그룹에서 누락된 물리 볼륨을 제거합니다.

```
# vgreduce -f vgold
# vgreduce -f vgnew
```
11. 다음과 같이 이전 볼륨 그룹에 대한 쿼럼 검사를 활성화합니다.

```
# vgchange -a y -q y /dev/vgold
```

완료 시 원래 볼륨 그룹에는 세 개의 논리 볼륨(lvol1, lvol2, lvol3)과 물리 볼륨 /dev/disk/disk0 및 /dev/disk/disk1이 포함됩니다. 새 볼륨 그룹 vgnew에는 물리 볼륨 /dev/disk/disk2, /dev/disk/disk3, /dev/disk/disk4, /dev/disk/disk5에 걸쳐 세 개의 논리 볼륨(lvol4, lvol5, lvol6)이 포함됩니다.

볼륨 그룹 제거



팁: 볼륨 그룹을 제거하려면 vgremove를 실행하기 전에 볼륨 그룹에서 논리 볼륨과 물리 볼륨을 모두 제거해야 하므로 볼륨 그룹을 제거하기 보다는 내보내기가 더 쉽습니다. 또한 볼륨 그룹을 내보내면 LVM 정보가 디스크에 그대로 남아 있으므로 나중에 볼륨 그룹을 다시 가져올 경우 유용합니다. 볼륨 그룹을 내보내는 절차는 “볼륨 그룹 내보내기” (69 페이지)를 참조하십시오.

볼륨 그룹을 제거하려면 다음 단계를 수행합니다.

1. 모든 사용자 데이터를 백업합니다.
2. 볼륨 그룹에 있는 모든 논리 볼륨과 물리 볼륨의 이름을 찾습니다. 다음 명령을 입력합니다.

```
# vgdisplay -v /dev/vgmn
```
3. 해당 논리 볼륨이 사용 중이 아닌지 확인합니다. 이 작업을 수행하려면 볼륨 그룹의 논리 볼륨을 사용하는 응용 프로그램을 중지하고 볼륨 그룹에 들어 있는 파일 시스템을 마운트 해제해야 합니다.
 각 논리 볼륨에 대해 `fuser` 명령을 사용합니다.

```
# fuser -cu /dev/vgmn/lvoln
```
4. 다음과 같이 각 논리 볼륨을 제거합니다.

```
# lvremove /dev/vgmn/lvoln
```

 자세한 내용은 “논리 볼륨 제거” (69 페이지)를 참조하십시오.
5. 다음과 같이 물리 볼륨 중 하나를 제외하고 모두 제거합니다.

```
# vgreduce /dev/vgmn /dev/disk/diskn
```

 자세한 내용은 “볼륨 그룹에서 디스크 제거” (62 페이지)를 참조하십시오.
6. `vgremove` 명령을 사용하여 다음과 같이 볼륨 그룹을 제거합니다.

```
# vgrename vgmn
```

미러 논리 볼륨 백업



참고: 미러링에는 선택적 제품인 HP MirrorDisk/UX가 필요합니다.

미러 논리 볼륨을 두 논리 볼륨으로 분리하여 다른 사본이 온라인 상태로 있는 동안 오프라인 사본에서 백업을 수행할 수 있습니다. 오프라인 사본에서 백업을 완료한 후 두 논리 볼륨을 다시 하나로 병합할 수 있습니다. 두 사본을 다시 동기화하기 위해 LVM에서는 사용 중이었던 사본의 변경 내용에 따라 오프라인 사본의 물리 확장 영역을 업데이트합니다.

HP SMH를 사용하여 논리 볼륨을 분리하거나 병합할 수 있으며 `lvsplit` 및 `lvmerge` 명령을 사용할 수도 있습니다.

`lvsplit` 및 `lvmerge`를 사용하여 파일 시스템이 들어 있는 미러 논리 볼륨을 백업하려면 다음 단계를 수행합니다.

1. 다음과 같이 논리 볼륨 `/dev/vg00/lvol1`을 두 개의 개별 논리 볼륨으로 분리합니다.

```
# lvsplit /dev/vg00/lvol1
```

 이 명령은 새 논리 볼륨 `/dev/vg00/lvol1b`를 만듭니다. 원래 논리 볼륨 `/dev/vg00/lvol1`은 온라인 상태로 유지됩니다.
2. 다음과 같이 백업할 논리 볼륨에서 파일 시스템 일관성 검사를 수행합니다.

```
# fsck /dev/vg00/lvol1b
```

3. 다음과 같이 파일 시스템을 마운트합니다.

```
# mkdir /backup_dir
# mount /dev/vg00/lvol1b /backup_dir
```

4. 선택한 유틸리티를 사용하여 백업을 수행합니다.

5. 다음과 같이 파일 시스템을 마운트 해제합니다.

```
# umount /backup_dir
```

6. 다음과 같이 분리된 논리 볼륨을 다시 원래 논리 볼륨과 병합합니다.

```
# lvmerge /dev/vg00/lvol1b /dev/vg00/lvol1
```

볼륨 그룹 구성 백업 및 복원

다음과 같이 구성이 변경될 때마다 볼륨 그룹 구성 정보를 저장해야 합니다.

- 볼륨 그룹에서 디스크 추가 또는 제거
- 루트 볼륨 그룹에서 디스크 변경
- 논리 볼륨 만들기 또는 제거
- 논리 볼륨 확장 및 축소

주어진 디스크의 알려진 위치에서 시작하고 끝나는 고정된 디스크 파티션 또는 파티션 되지 않은 디스크와 달리 각 볼륨 그룹 구성은 고유하고 변경되며 여러 디스크의 공간을 사용할 수 있습니다.

볼륨 그룹 구성을 백업하면 디스크 장애 또는 LVM 구성 정보 손상으로 인해 손상되거나 손실된 LVM 구성을 복원할 수 있습니다.

`vgcfgbackup` 명령은 볼륨 그룹 구성이 들어 있는 백업 파일을 만들거나 업데이트하지 만 **논리 볼륨 내의 데이터를 백업하지 않습니다**. 백업 절차를 간단하게 하기 위해 다음 명령을 사용하여 구성을 변경할 때마다 `vgcfgbackup`이 자동으로 호출됩니다.

<code>lvchange</code>	<code>lvcreate</code>	<code>lvextend</code>	<code>lvlnboot</code>	<code>lvmerge</code>
<code>lvreduce</code>	<code>lvremove</code>	<code>lvrmboot</code>	<code>lvsplit</code>	<code>vgcreate</code>
<code>pvchange</code>	<code>pvmove</code>	<code>vgextend</code>	<code>vgmodify</code>	<code>vgreduce</code>

`vgcfgbackup`을 사용하여 이전에 백업된 LVM 구성 정보를 표시하거나 `vgcfgrestore`를 사용하여 복원할 수 있습니다.

기본적으로 `vgcfgbackup`은 볼륨 그룹의 구성을

`/etc/lvmconf/volume_group_name.conf` 파일에 저장합니다.

명령줄에서 `vgcfgbackup`을 실행하여 지정한 디렉토리에 백업 파일을 저장할 수도 있습니다. 이렇게 하려면 우선 `vgdisplay`에 `-v` 옵션을 실행하여 볼륨 그룹의 모든 논리 볼륨이 `available/syncd`로 표시되는지 확인한 다음 이렇게 표시되면 다음 명령을 실행합니다.

```
# vgcfgbackup -f pathname/filename volume_group_name
```


기본이 아닌 볼륨 그룹 구성 파일을 사용하는 경우 해당 위치를 기록하여 보관하십시오. 명령 옵션에 대한 자세한 내용은 **vgcfgbackup(1M)**을 참조하십시오. 복구 중에 필요할 수 있는 루트 볼륨 그룹의 백업이 루트 파일 시스템에 있는지 확인합니다.

vgcfgrestore를 실행하려면 물리 볼륨을 분리해야 합니다. 물리 볼륨의 모든 데이터가 미러링되고 미러 사본이 최신 상태로 사용 가능하면 **pvchange**를 사용하여 물리 볼륨을 임시로 분리하고 **vgcfgrestore**를 수행한 후 물리 볼륨을 다시 연결할 수 있습니다. 예를 들어 볼륨 그룹 **/dev/vgsales**의 디스크 **/dev/disk/disk5**에 대한 볼륨 그룹 구성 데이터를 복원하려면 다음 명령을 입력합니다.

```
# pvchange -a n /dev/disk/disk5
# vgcfgrestore -n /dev/vgsales /dev/rdisk/disk5
# pvchange -a y /dev/disk/disk5
```

물리 볼륨이 미러링되지 않거나 미러 사본이 최신 상태로 사용 가능하지 않으면 **vgchange**를 사용하여 볼륨 그룹을 비활성화하고 **vgcfgrestore**를 수행한 후 다음과 같이 볼륨 그룹을 활성화해야 합니다.

```
# vgchange -a n /dev/vgsales
# vgcfgrestore -n /dev/vgsales /dev/rdisk/disk5
# vgchange -a y /dev/vgsales
```

이러한 예제에서는 LVM 구성이 **/etc/lvmconf/vgsales.conf**의 기본 백업 위치에서 디스크로 복원됩니다.

명령 옵션에 대한 자세한 내용은 **vgcfgrestore(1M)**를 참조하십시오.

디스크 이동 및 재구성

이 절에서는 다음 내용에 대해 설명합니다.

- “시스템 내에서 디스크 이동” (82 페이지)
- “시스템 간 디스크 이동” (83 페이지)
- “다른 물리 볼륨으로 데이터 이동” (84 페이지)
- “물리 볼륨 특성 수정” (87 페이지)
- “물리 볼륨에 대한 경로 비활성화” (94 페이지)
- “대체 부팅 디스크 만들기” (95 페이지)
- “부팅 디스크 미러링” (99 페이지)
- “HP 9000 서버에서 부팅 디스크 미러링” (99 페이지)
- “HP Integrity 서버에서 부팅 디스크 미러링” (101 페이지)

다음 작업을 수행해야 할 수 있습니다.

- 볼륨 그룹의 디스크를 시스템의 다른 하드웨어 위치로 이동합니다.

- 한 시스템에서 다른 시스템으로 디스크의 전체 볼륨 그룹을 이동합니다.



주의: 루트 볼륨 그룹의 일부인 디스크를 이동하는 것은 좋지 않습니다. 자세한 내용은 **주변 장치를 위한 HP-UX 구성**을 참조하십시오.

/etc/lvmtab 및 and /etc/lvmtab_p 파일에는 시스템에서 볼륨 그룹에 대한 LVM 디스크 매핑 정보 즉, 볼륨 그룹 이름 및 볼륨 그룹에 들어 있는 물리 볼륨 목록이 들어 있습니다. 이전 작업 중 하나를 수행할 때 이러한 구성 파일은 디스크에 대한 새 하드웨어 위치 및 장치 파일을 반영하도록 변경되어야 합니다. 그러나 이러한 파일은 텍스트 파일이 아니므로 직접 편집할 수 없습니다. 대신 LVM 구성 파일의 구성 변경 내용을 기록하는 vgexport 및 vgimport를 사용하여 볼륨 그룹을 다시 구성해야 합니다.

시스템 내에서 디스크 이동

볼륨 그룹의 디스크를 시스템의 다른 하드웨어 위치로 이동할 수 있는 두 가지 절차가 있습니다. 물리 볼륨에 대해 영구 또는 기존 장치 파일을 사용하는지 여부에 따라 절차를 선택합니다. 장치 파일 유형은 “기존 장치 파일과 영구 장치 파일 비교” (25 페이지)에 설명되어 있습니다.

영구 장치 파일을 사용하여 LVM 구성

LVM 구성에서 영구 장치 파일을 사용하는 경우 다음 단계를 수행합니다.

1. 볼륨 그룹의 데이터와 볼륨 그룹 구성에 대한 최신 백업이 있는지 확인합니다.
2. 다음 명령을 입력하여 볼륨 그룹을 비활성화합니다.

```
# vgchange -a n /dev/vgmn
```

3. 물리적으로 디스크를 원하는 새 위치로 이동합니다.
4. 다음과 같이 볼륨 그룹을 활성화합니다.

```
# vgchange -a y /dev/vgmn
```

기존 장치 파일을 사용하여 LVM 구성

물리 장치의 하드웨어 경로가 변경될 때 기존 장치 파일의 이름이 변경됩니다. 따라서 새로운 기존 장치 파일을 사용할 볼륨 그룹을 내보내고 가져와 LVM 구성을 업데이트합니다. 다음 단계를 수행하십시오.

1. 볼륨 그룹의 데이터와 볼륨 그룹 구성에 대한 최신 백업이 있는지 확인합니다.
2. 다음과 같이 볼륨 그룹을 비활성화합니다.

```
# vgchange -a n /dev/vgmn
```

3. 볼륨 그룹에 동일한 보조 번호를 유지하려면 다음과 같이 볼륨 그룹의 group 파일을 검토합니다.

```
# ls -l /dev/vgmn/group
```

```
crw-r--r-- 1 root sys 64 0x010000 Mar 28 2004 /dev/vgmn/group
```

이 예제에서 볼륨 그룹 주 번호는 64이고 보조 번호는 0x010000입니다.

4. 다음 명령을 입력하여 LVM 구성 파일에서 볼륨 그룹 장치 파일 및 해당 항목을 제거합니다.

```
# vgexport -v -s -m /tmp/vgmn.map /dev/vgmn
```
5. 물리적으로 디스크를 원하는 새 위치로 이동합니다.
6. 새 위치를 보려면 다음 명령을 입력합니다.

```
# vgscan -v
```
7. 2008년 3월 이전의 HP-UX 릴리즈를 사용하고 있거나 볼륨 그룹 장치 파일의 보조 번호를 유지하려면 “볼륨 그룹 장치 파일 만들기” (60 페이지)의 절차를 사용하여 만듭니다.

이 예제에서 group 파일의 주 번호는 64이고 보조 번호는 0x01000000이므로 다음 명령을 입력합니다.

```
# mkdir /dev/vgmn
# mknod /dev/vgmn/group c 64 0x010000
```
8. 다음과 같이 vgimport 명령을 사용하여 LVM 구성 파일에 볼륨 그룹 항목을 다시 추가합니다.

```
# vgimport -v -s -m /tmp/vgmn.map /dev/vgmn
```
9. 다음과 같이 새로 가져온 볼륨 그룹을 활성화합니다.

```
# vgchange -a y /dev/vgmn
```
10. 다음과 같이 볼륨 그룹 구성을 백업합니다.

```
# vgcfgbackup /dev/vgmn
```

시스템 간 디스크 이동

볼륨 그룹의 디스크를 다른 시스템의 다른 하드웨어 위치로 이동하려면 한 시스템에서 볼륨 그룹을 내보내고 디스크를 다른 시스템으로 물리적으로 이동한 후 볼륨 그룹을 해당 위치로 가져옵니다. 볼륨을 내보내고 가져오는 절차는 “볼륨 그룹 내보내기” (69 페이지) 및 “볼륨 그룹 가져오기” (70 페이지)에 설명되어 있습니다. 다음 예제에서는 이러한 절차를 설명합니다.



참고: 볼륨 그룹에 다중 경로 지정된 디스크가 들어 있으면 “볼륨 그룹 가져오기” (70 페이지) 아래 참고를 참조하십시오.

볼륨 그룹 /dev/vg_planning의 디스크 세 개를 다른 시스템으로 이동하려면 다음 단계를 수행합니다.

1. 논리 볼륨에 파일 시스템이 들어 있는 경우 파일 시스템을 마운트 해제합니다. 논리 볼륨이 보조 스왑으로 사용되는 경우 스왑을 비활성화하고 시스템을 다시 부팅해야 합니다. 보조 스왑에 대한 자세한 내용은 HP-UX 시스템 관리 설명서: 구성 관리를 참조하십시오.
2. 다음과 같이 볼륨 그룹 및 관련 논리 볼륨을 사용자가 사용할 수 없도록 지정합니다.

```
# vgchange -a n /dev/vg_planning
```

3. 다음 명령을 사용하여 LVM 구성 파일에서 볼륨 그룹 정보를 미리 봅니다.

```
# vgexport -p -v -s -m /tmp/vg_planning.map /dev/vg_planning
```

 -m 옵션을 사용하면 LVM 구성 파일에서 제거된 정보를 보유할 맵 파일의 이름을 지정할 수 있습니다. 맵 파일에는 볼륨 그룹에 있는 모든 논리 그룹의 이름이 들어 있습니다. 새로운 시스템에 볼륨 그룹을 설치할 때 이 맵 파일을 사용합니다.
4. 미리 본 내용이 만족스러우면 다음과 같이 볼륨 그룹 정보를 제거합니다.

```
# vgexport -v -s -m /tmp/vg_planning.map /dev/vg_planning
```

 vgexport 명령은 시스템에서 볼륨 그룹을 제거하고 /tmp/vg_planning.map 파일을 만듭니다.
5. 디스크를 새 시스템에 연결하고 /tmp/vg_planning.map 파일을 새 시스템으로 복사합니다.
6. 2008년 3월 이전의 HP-UX 릴리즈를 사용하고 있는 경우 “볼륨 그룹 장치 파일 만들기” (60 페이지)의 절차를 사용하여 볼륨 그룹 장치 파일을 만듭니다.
7. 디스크에 대한 장치 파일 정보를 가져오려면 ioscan 명령을 실행합니다.

```
# ioscan -funN -C disk
```
8. 가져오기 작업을 미리 보려면 -p 옵션과 함께 vgimport 명령을 실행합니다.

```
# vgimport -p -N -v -s -m /tmp/vg_planning.map /dev/vg_planning
```
9. 볼륨 그룹을 가져오려면 다음과 같이 -p 옵션 없이 vgimport를 실행합니다.

```
# vgimport -N -v -s -m /tmp/vg_planning.map /dev/vg_planning
```
10. 다음과 같이 새로 가져온 볼륨 그룹을 활성화합니다.

```
# vgchange -a y /dev/vg_planning
```

다른 물리 볼륨으로 데이터 이동

pvmove 명령을 사용하여 논리 볼륨에 들어 있는 데이터를 한 디스크에서 다른 디스크로 이동하거나 볼륨 그룹 내에서 디스크 간에 데이터를 이동할 수 있습니다.

예를 들어 특정 논리 볼륨의 데이터를 한 디스크에서 다른 디스크로 이동하여 첫 번째 디스크에서 비운 공간을 다른 용도로 사용할 수 있습니다. 논리 볼륨 /dev/vg01/markets의 데이터를 디스크 /dev/disk/disk4에서 디스크 /dev/disk/disk7로 이동하려면 다음을 입력합니다.

```
# pvmove -n /dev/vg01/markets /dev/disk/disk4 /dev/disk/disk7
```

반대로 관련된 논리 볼륨에 관계없이 한 디스크에 들어 있는 모든 데이터를 같은 볼륨 그룹 내에 있는 다른 디스크로 이동할 수 있습니다. 예를 들어 이 작업을 수행하여 볼륨 그룹에서 디스크를 제거합니다. pvmove를 사용하여 지정된 다른 디스크로 데이터를 이동하거나 미래링 할당 정책에 맞게 LVM이 볼륨 그룹에서 사용할 수 있는 적절한 공간으로 데이터를 이동하도록 할 수 있습니다.

디스크 /dev/dsk/disk3의 모든 데이터를 이동하여 대상 디스크 /dev/disk/disk5에 다시 배치하려면 다음 명령을 입력합니다.

```
# pvmove /dev/disk/disk3 /dev/disk/disk5
```

디스크 `/dev/disk/disk3`의 모든 데이터를 이동하여 LVM이 볼륨 그룹에서 사용할 수 있는 공간으로 데이터를 전달하도록 하려면 다음 명령을 입력합니다.

```
# pvmove /dev/disk/disk3
```

이전 경우에서 대상 디스크에 공간이 없으면 `pvmove` 명령이 실패합니다.



참고: `pvmove` 명령은 원자성 작업이 아니므로 데이터 확장 영역을 확장 영역별로 이동합니다. `pvmove`가 시스템 고장이나 `kill -9`에 의해 비정상적으로 종료되면 볼륨 그룹은 이동 중인 확장 영역의 추가 의사 미리 사본을 표시하는 일관되지 않은 구성으로 남을 수 있습니다. 영향을 받는 각 논리 볼륨에서 `lvreduce` 명령과 `-m` 옵션을 함께 사용하여 추가 미리 사본을 제거할 수 있습니다. 디스크를 지정할 필요가 없습니다.

스페어 디스크 만들기



참고: 디스크 스페어링을 사용하려면 선택적 제품인 HP MirrorDisk/UX가 필요합니다.

버전 2.x 볼륨 그룹은 디스크 스페어링을 지원하지 않습니다.

디스크 장애에 대비하여 보호할 볼륨 그룹에 스페어 물리 볼륨을 구성하려면 실제로 디스크 장애가 발생하기 전에 다음 단계를 수행합니다.

1. `pvcreate` 명령을 사용하여 디스크를 LVM 디스크로 초기화합니다.



참고: 스페어 물리 볼륨에는 부팅 정보가 들어갈 수 없으므로 `-B` 옵션은 사용하지 마십시오.

```
# pvcreate /dev/rdisk/disk3
```

2. 다음과 같이 볼륨 그룹이 활성화되었는지 확인합니다.

```
# vgchange -a y /dev/vg01
```

3. `vgextend` 명령에 `-z y`를 사용하여 하나 이상의 물리 볼륨을 볼륨 그룹의 스페어 물리 볼륨으로 지정합니다. 예를 들면 다음과 같습니다.

```
# vgextend -z y /dev/vg01 /dev/disk/disk3
```

또는 `pvchange` 명령에 `-z y` 옵션을 사용하여 현재 할당된 확장 영역이 없는 물리 볼륨을 스페어 물리 볼륨으로 변경할 수 있습니다. 예를 들면 다음과 같습니다.

```
# pvchange -z y /dev/disk/disk3
```

디스크 스페어링에 대한 자세한 내용은 “[디스크 스페어링을 통해 디스크 중복 늘리기](#)” (42 페이지)를 참조하십시오.

스페어 디스크 복귀



참고: 디스크 스페어링을 사용하려면 선택적 제품인 HP MirrorDisk/UX가 필요합니다.

버전 2.x 볼륨 그룹은 디스크 스페어링을 지원하지 않습니다.

장애가 발생한 디스크가 복구되었거나 교체하기로 결정한 후에는 다음 단계를 따라 스페어 디스크를 복귀하고 이전의 대기 상태로 되돌립니다.

1. 새 디스크 또는 복구된 디스크에 물리적으로 연결합니다.
2. 다음과 같이 `vgcfgrestore`를 사용하여 다시 연결된 디스크로 LVM 구성을 복원합니다.

```
# vgcfgrestore -n /dev/vg01 /dev/rdisk/disk1
```

3. 다음과 같이 볼륨 그룹이 활성화되었는지 확인합니다.

```
# vgchange -a y /dev/vg01
```

4. 다음과 같이 교체된 디스크에 확장 영역의 할당이 허용되는지 확인합니다.

```
# pvchange -x y /dev/disk/disk1
```

5. pvmove 명령을 사용하여 스페어 볼륨에서 교체된 물리 볼륨으로 데이터를 이동합니다. 예를 들면 다음과 같습니다.

```
# pvmove /dev/disk/disk3 /dev/disk/disk1
```

스페어 디스크의 데이터를 원래 디스크 또는 교체 디스크로 되돌리고 스페어 디스크를 비어 있는 대기 디스크로 되돌립니다.

디스크 스페어링에 대한 자세한 내용은 “디스크 스페어링을 통해 디스크 중복 늘리기”(42 페이지)를 참조하십시오.

물리 볼륨 특성 수정



참고: 버전 2.x 볼륨 그룹은 부팅 가능한 물리 볼륨을 지원하지 않습니다.

vgmodify 명령을 사용하여 물리 볼륨의 변경 내용에 맞게 볼륨 그룹을 수정할 수 있습니다. 특히 물리 볼륨의 크기 변경을 인식하도록 볼륨 그룹을 조정하고 물리 볼륨 유형을 부팅 가능과 부팅 불가능 사이에서 변경할 수 있습니다.

크기 변경 인식

대개 디스크 어레이를 통해 LUN의 크기를 조정할 수 있습니다. LUN 크기를 늘릴 경우 다음 단계를 수행하여 추가 공간을 볼륨 그룹에 통합합니다.

1. 어레이에 대한 명령을 사용하여 LUN 크기를 늘립니다.
2. vgmodify를 실행하여 물리 볼륨 크기 변경을 감지합니다. 또한 모든 공간을 볼륨 그룹으로 사용할 수 있는지 여부를 보고합니다.
3. vgmodify가 물리 볼륨당 최대 물리 확장 영역 수(max_pe)가 너무 작아 새 크기를 수용할 수 없다고 보고하면 “볼륨 그룹 매개 변수 수정”(70 페이지)의 설명에 따라 vgmodify와 -t 및 -n 옵션을 함께 사용하여 max_pe의 새 값을 결정합니다.
4. vgmodify를 새 설정 및 -r 옵션과 함께 실행하여 값을 검토합니다.
5. 볼륨 그룹을 비활성화합니다.
6. max_pe의 새 값을 커밋하고 vgmodify를 -r 옵션 없이 실행하여 물리 볼륨 정보를 업데이트합니다.
7. 볼륨 그룹을 활성화합니다. vgdisplay와 pvdisplay 명령을 실행하여 증가한 공간이 사용 가능한지 확인합니다.



팁: >HP-UX 11i v3의 2008년 3월 릴리즈부터는 vgmodify에 -E 및 -a 옵션을 사용하여 볼륨 그룹을 다시 활성화하지 않고도 크기 변경 사항을 인식하여 수용할 수 있습니다. 자세한 내용은 vgmodify(1M)를 참조하십시오.

예를 들어 물리 볼륨 /dev/rdisk/disk6의 크기를 4GB에서 100000000KB로 늘리면 다음 단계를 수행합니다.

1. 디스크 어레이에 대한 명령을 사용하여 LUN 크기를 늘립니다.
2. `vgmodify`를 `-v` 및 `-r` 옵션과 함께 실행하여 디스크 크기가 변경되었는지 여부 및 물리 볼륨의 모든 공간을 사용할 수 있는지 여부를 확인합니다.

```
# vgmodify -v -r vg32
현재 볼륨 그룹 설정:
```

최대 LV	255
최대 PV	16
PV당 최대 PE	1016
PE 크기 (MB)	32
VGRA 크기 (KB)	176

/dev/rdisk/disk6 경고: 볼륨 그룹에 대한 PV당 최대 PE(1016) 가
이 PV에 비해 너무 작습니다(3051).
이 물리 볼륨에서 1016개의 PE만 사용 중입니다.

"/dev/rdisk/disk6" 크기가 4194304에서 1000000000kb로 변경되었습니다.

볼륨 그룹에 대한 업데이트가 필요합니다.

새 볼륨 그룹 설정:

최대 LV	255
최대 PV	16
PV당 최대 PE	1016
PE 크기 (MB)	32
VGRA 크기 (KB)	176

검토를 완료했습니다. 볼륨 그룹이 수정되지 않았습니다.

확장된 물리 볼륨은 3051개 물리 확장 영역의 공간을 모두 사용해야 하지만 현재 `max_pe` 값은 이 개수를 1016으로 제한합니다.

3. `max_pv` 및 `max_pe`의 최적 값을 결정하려면 다음과 같이 `-n`의 포함 여부에 관계 없이 `vgmodify -t`를 실행합니다.

```
# vgmodify -t vg32
현재 볼륨 그룹 설정:
```

최대 LV	255
최대 PV	16
PV당 최대 PE	1016
PE 크기 (MB)	32
VGRA 크기 (KB)	176
VGRA 공간 (KB) PE 번호 재지정 없음	896
VGRA 공간 (KB) PE 번호 재지정 낮음	32768

볼륨 그룹 최적화된 설정 (번호 재지정된 PE 없음):

max_pv (-p)	max_pe (-e)	디스크 크기 (Mb)
2	53756	1720193
3	35836	1146753
4	26876	860033
...		
28	3836	122753
30	3580	114561
32	3324	106369
35	3068	98177
38	2812	89985
...		
255	252	8065

표에서는 물리 확장 영역의 번호를 다시 매기지 않을 경우 max_pv가 35이하여야 증가한 물리 볼륨 크기를 충분히 수용할 수 있는 max_pe를 허용함을 보여 줍니다.

```
# vgmodify -v -t -n vg32
/dev/vg32에 대한 볼륨 그룹 구성이
/etc/lvmconf/vg32.conf에 저장되었습니다.
```

현재 볼륨 그룹 설정:

최대 LV	255
최대 PV	16
PV당 최대 PE	1016
PE 크기 (MB)	32
VGRA 크기 (KB)	176

모든 물리 볼륨의 VGRA 공간 (KB):

PV	현재	-n
/dev/rdisk/disk6	896	32768
/dev/rdisk/disk5	896	32768
요약	896	32768

볼륨 그룹 최적화된 설정 (PE가 낮게 번호 재지정됨):

max_pv (-p)	max_pe (-e)	디스크 크기 (Mb)
61	65535	2097152
62	65532	2097056
63	64252	2056096
...		
251	16124	516000
252	16048	513568
255	15868	507808

표에서는 물리 확장 영역의 번호가 다시 매겨지면 max_pv의 모든 값이 증가한 물리 볼륨 크기를 충분히 수용할 수 있는 max_pe를 허용함을 보여 줍니다.

이 예제의 경우 max_pv로 10을 선택하여 max_pe 값으로 10748을 허용합니다.

- 다음과 같이 vgmodify에 -r 옵션을 사용하여 변경 내용을 미리 봅니다.

```
# vgmodify -p 10 -e 10748 -r vg32
```

현재 볼륨 그룹 설정:

최대 LV	255
최대 PV	16
PV당 최대 PE	1016
PE 크기 (MB)	32
VGRA 크기 (KB)	176

현재 값과 새 볼륨 그룹 매개 변수가 다릅니다.

"/dev/rdisk/disk6" 크기가 4194304에서 100000000kb로 변경되었습니다.

볼륨 그룹에 대한 업데이트가 필요합니다.

새 볼륨 그룹 설정:

최대 LV	255
최대 PV	10
PV당 최대 PE	10748
PE 크기 (MB)	32
VGRA 크기 (KB)	896

검토를 완료했습니다. 볼륨 그룹이 수정되지 않았습니다.

- 다음과 같이 볼륨 그룹을 비활성화합니다.

```
# vgchange -a n vg32
볼륨 그룹 "vg32"을(를) 성공적으로 변경했습니다.
```

6. 다음과 같이 새 값을 커밋합니다.

```
# vgmodify -p 10 -e 10748 vg32
현재 볼륨 그룹 설정:
```

최대 LV	255
최대 PV	16
PV당 최대 PE	1016
PE 크기(MB)	32
VGRA 크기(KB)	176

현재 값과 새 볼륨 그룹 매개 변수가 다릅니다.
 "/dev/rdisk/disk6" 크기가 4194304에서 100000000kb로 변경되었습니다.
 볼륨 그룹에 대한 업데이트가 필요합니다.

새 볼륨 그룹 설정:

최대 LV	255
최대 PV	10
PV당 최대 PE	10748
PE 크기(MB)	32
VGRA 크기(KB)	896

"vg32"에 대한 새 볼륨 그룹 구성이
 "/etc/lvmconf/vg32.conf"에 저장되었습니다.
 "vg32"에 대한 이전 볼륨 그룹 구성이
 "/etc/lvmconf/vg32.conf.old"에 저장되었습니다.
 모든 물리 볼륨에 써서 수정을 시작하는 중입니다.
 "/etc/lvmconf/vg32.conf"에서 모든 물리 볼륨으로
 설정을 적용 중입니다.
 수정 프로세스를 완료했습니다.
 "vg32"에 대한 새 볼륨 그룹 구성이
 "/etc/lvmconf/vg32.conf.old"에 저장되었습니다.
 볼륨 그룹 "vg32"을(를) 성공적으로 변경했습니다.

7. 다음 명령을 입력하여 볼륨 그룹을 활성화하고 변경 내용을 확인합니다.

```
# vgchange -a y vg32
활성화된 볼륨 그룹
볼륨 그룹 "vg32"을(를) 성공적으로 변경했습니다.
```

```
# vgdisplay vg32
--- 볼륨 그룹 ---
```

VG 이름	/dev/vg32
VG 쓰기 권한	읽기/쓰기
VG 상태	사용 가능
최대 LV	255
현재 LV	0
열린 LV	0
최대 PV	10
열린 PV	2
활성 PV	2
PV당 최대 PE	10748
VGDA	4
PE 크기(MB)	32
총 PE	3119
할당된 PE	0

사용 가능한 PE	3119
총 PVG	0
총 스페어 PV	0
사용 중인 총 스페어 PV	0
VG 버전	1.0



주의: 이 절차는 물리 볼륨의 크기가 줄어든 때에도 사용 가능합니다. 그러나 제한이 있습니다.

- 데이터 손상을 방지하려면 `vgmodify`를 통해 볼륨 그룹을 성공적으로 변경한 **이후**에 디스크 어레이에 있는 LUN의 크기를 축소시켜야 합니다.
- 축소시키기 전에 볼륨 그룹을 비활성화해야 합니다. 볼륨 그룹이 활성화된 상태에서 LUN의 크기를 축소시키면 LVM은 물리 볼륨을 사용할 수 없다고 표시합니다.
- 물리 볼륨에서 할당된 물리 확장 영역이 대상 크기를 초과할 경우 `vgmodify`는 물리 볼륨을 변경하지 않고 오류 메시지를 출력한 다음 종료합니다. 이 경우에 LUN을 원래의 크기로 복원할 준비를 하여 동일한 디스크 공간이 할당되는지 확인합니다.

물리 볼륨 부팅 유형 변경



참고: `vgmodify` 명령은 버전 2.x 볼륨 그룹을 지원하지 않습니다.

물리 볼륨이 LVM용으로 초기화되면 부팅 가능 또는 부팅 불가능으로 설정할 수 있습니다. 부팅 가능한 물리 볼륨을 사용하려면 LVM 메타데이터에 부팅 유틸리티와 정보에 대한 추가 공간이 필요합니다. 물리 볼륨이 실수로 부팅 가능으로 초기화된 경우 디스크를 부팅 불가능 디스크로 변환하고 LVM 메타데이터 공간을 되찾을 수 있습니다.



주의: 부팅 볼륨 그룹에는 하나 이상의 부팅 가능한 물리 볼륨이 필요합니다. 부팅 볼륨 그룹의 물리 볼륨 전체를 부팅 불가능으로 변환하지 마십시오. 이렇게 변환하면 시스템이 부팅되지 않습니다.

디스크 유형을 부팅 가능에서 부팅 불가능으로 변경하려면 다음 단계를 수행합니다.

1. `vgcfgrestore`를 사용하여 볼륨 그룹에 부팅 가능 디스크가 들어 있는지 확인합니다.
2. `vgmodify`를 두 번 실행합니다. `-B n`과 함께 한 번 실행하고 이 옵션 없이 한 번 실행합니다. `max_pe` 및 `max_pv`의 사용 가능한 값을 비교합니다.
3. `max_pe` 및 `max_pv`의 새 값을 선택합니다. `vgmodify`를 새 설정 및 `-r` 옵션과 함께 실행하여 값을 검토합니다.
4. 볼륨 그룹을 비활성화합니다.
5. `-r` 옵션 없이 `vgmodify`를 실행하여 변경 내용을 커밋합니다.
6. 볼륨 그룹을 활성화합니다. `vgcfgrestore` 또는 `pvdiskdisplay` 명령을 실행하여 디스크 유형이 변경되었는지 확인합니다.

예를 들어 볼륨 그룹 `vg`의 부팅 가능한 디스크를 변환하려면 다음 단계를 수행합니다.

1. 다음과 같은 vg01의 물리 볼륨이 부팅 가능한지 확인합니다.

```
# vgcfgrestore -l -v -n vg01
"/etc/lvmconf/vg01.conf"의 볼륨 그룹 구성 정보
VG 이름 /dev/vg01
---- 물리 볼륨 : 1 ----
PV      유형      크기(kb)   시작(kb)   PVkey
c2t1d0 Bootable 35566480 2912       0

max_pv 16 max_pe 1085 max_lv 255
```

2. max_pe 및 max_pv의 값 중 사용 가능한 값을 확인하려면 다음 명령을 실행합니다.

```
# vgmodify -t -B n vg01 /dev/rdsdsk/c2t1d0
현재 볼륨 그룹 설정:
```

	최대 LV	255
	최대 PV	16
PV당 최대 PE	1085	
PE 크기(MB)	32	
VGRA 크기(KB)	208	
VGRA 공간(KB) PE 번호 재지정 없음	2784	
VGRA 공간(KB) PE 번호 재지정 낮음	32768	

```
볼륨 그룹 최적화된 설정(번호 재지정된 PE 없음):
max_pv(-p) max_pe(-e) 디스크 크기(Mb)
5           65535      2097122
6           56828      1818498
...
255         1276       40834
```

디스크가 부팅 불가능으로 설정된 경우의 값과 그렇지 않은 경우의 값을 비교합니다. 다음 명령을 입력합니다.

```
# vgmodify -t vg01
현재 볼륨 그룹 설정:
```

	최대 LV	255
	최대 PV	16
PV당 최대 PE	1085	
PE 크기(MB)	32	
VGRA 크기(KB)	208	
VGRA 공간(KB) PE 번호 재지정 없음	768	
VGRA 공간(KB) PE 번호 재지정 낮음	768	

```
볼륨 그룹 최적화된 설정(번호 재지정된 PE 없음):
max_pv(-p) max_pe(-e) 디스크 크기(Mb)
1           65535      2097120
2           45820      1466240
...
255         252       8064
```

디스크 유형을 변경하면 사용 가능한 VGRA 공간이 768KB에서 2784KB(물리 확장 영역의 번호가 다시 매겨지지 않는 경우) 또는 32768KB(물리 확장 영역의 번호가 다시 매겨지는 경우)로 증가합니다. 디스크 유형을 변경하면 더 큰 범위의 max_pv와 max_pe도 허용됩니다. 예를 들어 max_pv가 255이면 부팅 가능한 디스크는 8064MB의 디스크 크기만 수용할 수 있지만 부팅 불가능으로 변환한 후에는 40834MB의 디스크 크기를 수용할 수 있습니다.

3. 이 예제에서는 max_pv 값으로 6을 선택하여 max_pe 값으로 56828을 허용합니다.
다음 명령을 입력하여 변경 내용을 미리 봅니다.

```
# vgmodify -r -p 6 -e 56828 -B n vg01 /dev/rdsk/c2t1d0
```

현재 볼륨 그룹 설정:

최대 LV	255
최대 PV	16
PV당 최대 PE	1085
PE 크기 (MB)	32
VGRA 크기 (KB)	208

현재 값과 새 볼륨 그룹 매개 변수가 다릅니다.
볼륨 그룹에 대한 업데이트가 필요합니다.

새 볼륨 그룹 설정:

현재 볼륨 그룹 설정:

최대 LV	255
최대 PV	6
PV당 최대 PE	56828
PE 크기 (MB)	32
VGRA 크기 (KB)	2784

검토를 완료했습니다. 볼륨 그룹이 수정되지 않았습니다.

4. 다음과 같이 볼륨 그룹을 비활성화합니다.

```
# vgchange -a n vg01
```

볼륨 그룹 "vg01"을(를) 성공적으로 변경했습니다.

5. 다음과 같이 새 값을 커밋합니다.

```
# vgmodify -p 6 -e 56828 -B n vg01 /dev/rdsk/c2t1d0
```

현재 볼륨 그룹 설정:

최대 LV	255
최대 PV	16
PV당 최대 PE	1085
PE 크기 (MB)	32
VGRA 크기 (KB)	208

현재 값과 새 볼륨 그룹 매개 변수가 다릅니다.
볼륨 그룹에 대한 업데이트가 필요합니다.

새 볼륨 그룹 설정:

현재 볼륨 그룹 설정:

최대 LV	255
최대 PV	6
PV당 최대 PE	56828
PE 크기 (MB)	32
VGRA 크기 (KB)	2784

"vg01"에 대한 새 볼륨 그룹 구성이
"/etc/lvmconf/vg01.conf"에 저장되었습니다.
"vg01"에 대한 이전 볼륨 그룹 구성이
"/etc/lvmconf/vg01.conf.old"에 저장되었습니다.

모든 물리 볼륨에 써서 수정을 시작하는 중입니다.

"/etc/lvmconf/vg01.conf"에서 모든 물리 볼륨으로
설정을 적용 중입니다.

수정 프로세스를 완료했습니다.

"vg01"에 대한 새 볼륨 그룹 구성이
"/etc/lvmconf/vg01.conf.old"에 저장되었습니다.
볼륨 그룹 "vg01"을(를) 성공적으로 변경했습니다.

6. 다음과 같이 볼륨 그룹을 활성화하고 변경 내용을 확인합니다.

```
# vgchange -a y vg01
활성화된 볼륨 그룹
볼륨 그룹 "vg01"을(를) 성공적으로 변경했습니다.

# vgcfgbackup vg01
/dev/vg01에 대한 볼륨 그룹 구성이
/etc/lvmconf/vg01.conf에 저장되었습니다.

# vgcfgrestore -l -v -n vg01
"/etc/lvmconf/vg01.conf"의 볼륨 그룹 구성 정보
VG 이름 /dev/vg01

---- 물리 볼륨 : 1 ----
PV          유형      크기(kb)   시작(kb)   PVkey
c2t1d0 Non-Boot 35566480   2912       0

max_pv 6 max_pe 56828 max_lv 255
```

물리 볼륨에 대한 경로 비활성화



중요: 이 절차에서는 LVM의 링크 사용만 비활성화합니다. pvchange 명령은 진단이나 응용 프로그램이 물리 볼륨에 액세스하는 것을 허용합니다.

기본적으로, 대용량 저장소 스택에서는 LVM에서 구성된 경로와 상관 없이 사용 가능한 모든 경로를 사용하여 물리 볼륨에 액세스합니다. LVM에서 경로를 비활성화하더라도 기본 다중 경로 지정이 그 경로를 사용하는 것을 막지는 않습니다. 한 경로상의 I/O를 비활성화하거나 기본 다중 경로 지정을 비활성화하려면 scsimgr 명령을 사용합니다.

pvchange 명령을 사용하여 LVM의 물리 볼륨에 대한 물리 경로 중 하나 또는 모두를 일시적으로 비활성화할 수 있습니다. 경로를 비활성화(링크 분리라고도 함)하면 LVM에서 장치에 대한 해당 경로를 닫고 장치 사용을 중지합니다. 이렇게 하면 I/O 카드에서 진단을 실행하거나, I/O 카드를 교체하거나, 물리 볼륨이 포함된 디스크를 교체하는 경우와 같이 링크가 유휴 상태에 있도록 하는 경우에 편리합니다.

물리 볼륨에 대한 링크 분리는 영구 작업이 아닌 임시 작업입니다. 볼륨 그룹에서 링크나 물리 볼륨을 영구적으로 제거하려면 “볼륨 그룹에서 디스크 제거” (62 페이지)에 설명된 대로 vgreduce를 대신 사용합니다.

물리 볼륨에 대한 링크를 분리하려면 pvchange에 -a 옵션을 사용합니다. 예를 들어 /dev/disk/disk33 장치를 통해 링크를 비활성화하려면 다음 명령을 입력합니다.

```
# pvchange -a n /dev/disk/disk33
```

다중 경로 지정된 디스크에 대해 LVM의 대체 링크를 사용할 경우 각 링크는 서로 다른 기존 장치 파일을 사용합니다. 이 경우 물리 볼륨에 대한 모든 링크를 분리하려면 -a 옵션에 대한 인수로 N을 사용합니다.

```
# pvchange -a N /dev/dsk/c5t0d0
```

물리 볼륨에 대한 하나 이상의 링크를 분리해도 LVM에서 해당 볼륨의 사용을 완전히 중지하지는 않습니다. 분리된 링크가 장치에 대한 기본 경로인 경우 LVM은 해당 장치에 사용할 수 있는 대체 링크를 사용하여 시작됩니다. 물리 볼륨에 대한 링크가 모두 분리되는 경우에만 LVM에서 해당 물리 볼륨의 사용을 중지합니다.

장치에 대한 모든 링크를 분리하면 볼륨 그룹에서 연관된 물리 볼륨을 사용할 수 없습니다. 링크는 계속 볼륨 그룹과 연결되어 있지만 링크가 다시 연결되기 전까지는 LVM에서 물리 볼륨으로 I/O 요청을 보내지 않습니다. 따라서 해당 물리 볼륨의 데이터를 일시적으로 사용할 수 없게 됩니다. 그러므로 장치를 분리하여 사용할 수 없게 하기 전에 미리하여 해당 데이터의 가용성 요구 사항을 충족시켜야 합니다.

링크를 분리해도 스페어링이 비활성화되지는 않습니다. 따라서 물리 볼륨에 대한 링크를 모두 분리하더라도 볼륨 그룹에서 적당한 스페어 물리 볼륨을 사용할 수 있는 경우 LVM은 이 볼륨을 사용하여 분리된 디스크를 다시 구성합니다. 자세한 내용은 “**디스크 스페어링을 통해 디스크 중복 늘리기**” (42 페이지)를 참조하십시오.

vgdisplay에 -v 옵션을 사용하면 물리 볼륨에 대한 모든 링크의 LVM 상태를 볼 수 있습니다.

물리 볼륨에 대한 분리된 링크를 복원 또는 **다시 연결**하면 해당 링크를 볼륨 그룹에 사용할 수 있게 됩니다. LVM에서 필요한 대로 해당 링크를 사용하여 디스크에 액세스할 수 있게 됩니다.

물리 볼륨에 대한 특정 경로를 다시 연결하려면 pvchange 명령에 -a 옵션을 사용합니다. 예를 들어 다음 명령을 입력합니다.

```
# pvchange -a y /dev/dsk/c5t0d0
```

물리 볼륨에 대한 링크 분리는 임시 작업이므로 부팅 시나 다음과 같은 명시적 vgchange 명령을 사용하여 볼륨 그룹이 활성화되는 경우 볼륨 그룹에서 분리된 링크가 모두 다시 연결됩니다.

```
# vgchange -a y /dev/vg02
```

대체 부팅 디스크 만들기



참고: 버전 2.x 볼륨 그룹은 부팅 가능한 물리 볼륨을 지원하지 않습니다. 버전 2.x 볼륨 그룹에서는 대체 부팅 디스크를 만들 수 없습니다.

비LVM 디스크를 사용하는 경우에는 단일 루트 디스크에 시스템 파일, 기본 스왑 및 덤프뿐만 아니라 부팅에 필요한 모든 속성이 포함됩니다. LVM을 사용하면 단일 루트 디스크는 디스크 풀인 **루트 볼륨 그룹**으로 대체됩니다. 여기에는 모든 동일한 요소가 포함되지만 루트 논리 볼륨, 부팅 논리 볼륨, 스왑 논리 볼륨 및 하나 이상의 덤프 논리 볼륨을 허용합니다. 이러한 논리 볼륨은 연속적이어야 합니다. 즉, 하나의 디스크에 포함되어야 하며 손상된 블록 배치가 비활성화되어 있어야 합니다. 다른 비연속 논리 볼륨은 사용자 데이터에 사용할 수 있습니다. 스왑 및 덤프 장치와 해당 구성에 대한 자세한 내용은 **HP-UX 시스템 관리 설명서: 구성 관리**를 참조하십시오.

루트 논리 볼륨에는 운영 체제 소프트웨어 및 루트 파일 시스템(/)이 포함되어 있습니다. 부팅 논리 볼륨에는 부팅 파일 시스템(/stand)이 포함됩니다. 루트 및 부팅 논리 볼

를 하나의 논리 볼륨으로 결합하거나 별도로 유지할 수 있습니다. 하나의 결합된 루트-부팅 논리 볼륨을 사용하든지 개별 루트 및 부팅 논리 볼륨을 사용하든지 시스템을 부팅하는 데 사용되는 논리 볼륨은 해당 물리 볼륨의 첫 번째 논리 볼륨이어야 합니다. 시스템을 유지 관리 모드에서 부팅하려면 물리 확장 영역 0000에서 시작해야 합니다.

새로 HP-UX 시스템을 설치하고 LVM 구성을 선택하면 개별 루트(/dev/vg00/lvol3) 및 부팅(/dev/vg00/lvol1) 논리 볼륨처럼 루트 볼륨 그룹이 자동으로 구성됩니다 (/dev/vg00). 현재 결합된 루트 및 부팅 논리 볼륨이 있고 개별 루트 및 부팅 논리 볼륨으로 다시 구성하려면 부팅 논리 볼륨을 만든 후 `lvlnboot` 명령과 `-b` 옵션을 함께 사용하여 부팅 논리 볼륨을 시스템에 정의해야 합니다. 다음에 시스템을 부팅하면 변경 사항이 적용됩니다.

스왑 논리 볼륨은 시스템의 기본 스왑 영역이며 일반적으로 덤프에 사용됩니다. 스왑 논리 볼륨은 루트 논리 볼륨과 같은 물리 디스크에 있는 경우도 많습니다. 그러나 루트 논리 볼륨이 아닌 다른 물리 디스크에 스왑 논리 볼륨 및 덤프 논리 볼륨을 구성할 수 있습니다.

여러 디스크로 루트 볼륨 그룹을 만드는 경우 `lvextend` 명령을 사용하여 부팅, 루트 및 기본 스왑 논리 볼륨을 부팅 디스크에 배치합니다.



팁: 루트 논리 볼륨에 공간이 필요한 경우 `pvmove`를 사용하여 기존 논리 볼륨에서 다른 디스크로 데이터를 이동할 수 있습니다. 자세한 내용은 “다른 물리 볼륨으로 데이터 이동” (84 페이지)을 참조하십시오.

대체 부팅 디스크를 사용하여 새 루트 볼륨 그룹을 만들려면 다음 단계를 수행합니다.

1. 부팅 가능한 물리 볼륨을 만듭니다.
 - a. HP Integrity 서버인 경우 “HP Integrity 서버에서 부팅 디스크 미러링” (101 페이지)의 설명에 따라 `idisk` 명령 및 파티션 설명 파일을 사용하여 디스크를 파티션하고 `insf`를 실행합니다.
 - b. `pvccreate`에 `-B` 옵션을 실행합니다. HP Integrity 서버에서는 HP-UX 파티션을 나타내는 장치 파일을 사용합니다.

```
# pvccreate -B /dev/rdisk/disk6_p2
```

HP 9000 서버에서는 전체 디스크에 대한 장치 파일을 사용합니다.

```
# pvccreate -B /dev/rdisk/disk6
```
2. 볼륨 그룹에 대한 디렉토리를 만듭니다. 예를 들면 다음과 같습니다.

```
# mkdir /dev/vgroot
```
3. 다음과 같이 앞에서 설명한 디렉토리에 `group`이라는 장치 파일을 만듭니다.

```
# mknod /dev/vgroot/group c 64 0xnn0000
```
4. 다음과 같이 포함될 각 물리 볼륨을 지정하는 루트 볼륨 그룹을 만듭니다.

```
# vgcreate /dev/vgroot /dev/disk/disk6
```
5. 다음과 같이 부팅 영역에 부팅 유틸리티를 배치합니다.

- ```
mkboot /dev/rdisk/disk6
```
6. 다음과 같이 디스크 부팅 영역에 자동 부팅 파일을 추가합니다.
 

```
mkboot -a "hpux" /dev/rdisk/disk6
```
  7. 부팅 논리 볼륨을 만듭니다. 이름이 bootlv이고 크기가 512MB인 부팅 논리 볼륨을 만들려면 다음 명령을 입력합니다.
 

```
lvcreate -C y -r n -n bootlv /dev/vgroot
lvextend -L 512 /dev/vgroot/bootlv /dev/disk/disk6
```
  8. 기본 스왑 논리 볼륨을 만듭니다. 예를 들어, 이름이 swaplv이고 크기가 2GB인 기본 스왑 논리 볼륨을 부팅 논리 볼륨과 같은 디스크에 만들려면 다음 명령을 입력합니다.
 

```
lvcreate -C y -r n -n swaplv /dev/vgroot
lvextend -L 2048 /dev/vgroot/swaplv /dev/disk/disk6
```
  9. 루트 논리 볼륨을 만듭니다. 예를 들어, 이름이 rootlv이고 크기가 1GB인 루트 논리 볼륨을 만들려면 다음 명령을 입력합니다.
 

```
lvcreate -C y -r n -n rootlv /dev/vgroot
lvextend -L 1024 /dev/vgroot/rootlv /dev/disk/disk6
```
  10. 다음과 같이 bootlv가 부팅 논리 볼륨임을 지정합니다.
 

```
lvmboot -b /dev/vgroot/bootlv
```
  11. 다음과 같이 rootlv가 루트 논리 볼륨임을 지정합니다.
 

```
lvmboot -r /dev/vgroot/rootlv
```
  12. 다음과 같이 swaplv가 기본 스왑 논리 볼륨임을 지정합니다.
 

```
lvmboot -s /dev/vgroot/swaplv
```
  13. 다음과 같이 swaplv가 덤프용으로도 사용되도록 지정합니다.
 

```
lvmboot -d /dev/vgroot/swaplv
```
  14. 다음과 같이 구성을 확인합니다.
 

```
lvmboot -v /dev/vgroot
```

볼륨 그룹 /dev/vgroot에 대한 부팅 정의:  
 루트 볼륨 그룹에 속하는 물리 볼륨:  
                                   /dev/disk/disk6 -- 부팅 디스크

|            |     |                    |
|------------|-----|--------------------|
| 부팅: bootlv | 켜짐: | /dev/disk/disk6    |
| 루트: rootlv | 켜짐: | /dev/disk/disk6    |
| 스왑: swaplv | 켜짐: | /dev/disk/disk6    |
| 덤프: swaplv | 켜짐: | /dev/disk/disk6, 0 |
  15. 부팅 및 루트 논리 볼륨이 만들어졌으면 해당 볼륨에 대한 파일 시스템을 만듭니다. 예를 들면 다음과 같습니다.
 

```
mkfs -F hfs /dev/vgroot/rbootlv
mkfs -F vxfs /dev/vgroot/rrootlv
```



---

**참고:** HP Integrity 서버에서는 부팅 파일 시스템이 VxFS가 될 수 있습니다. 다음 명령을 입력합니다.

```
mkfs -F vxfs /dev/vgroot/rbootlv
```

---

## 부팅 디스크 미러링



**참고:** 미러링에는 선택적 제품인 HP MirrorDisk/UX가 필요합니다.

버전 2.x 볼륨 그룹에서는 부팅 디스크를 지원하지 않으므로 버전 2.x 볼륨 그룹에서는 부팅 디스크를 미러링할 수 없습니다.

루트, 부팅 및 기본 스왑 논리 볼륨의 미러 사본을 만든 후 이러한 기본 물리 볼륨에 장애가 발생하면 시스템에서 다른 디스크의 미러 사본을 사용하여 작업을 진행할 수 있습니다. 장애가 발생한 디스크가 다시 온라인 상태가 되면 시스템이 다시 부팅되지 않은 경우 자동으로 복구됩니다.

디스크가 다시 온라인 상태가 되기 전에 시스템이 다시 부팅되면 볼륨 그룹을 다시 활성화하여 볼륨 그룹의 디스크를 추적하는 LVM 데이터 구조를 업데이트합니다. 볼륨 그룹이 이미 활성화된 경우에도 `vgchange -a y`를 사용할 수 있습니다.

예를 들어 다음 명령을 입력하여 볼륨 그룹 `vg00`을 다시 활성화할 수 있습니다.

```
vgchange -a y /dev/vg00
```

결과적으로 LVM은 시스템이 다시 부팅된 후 온라인 상태가 된 디스크를 포함하여 볼륨 그룹 `vg00`에 있는 사용 가능한 모든 디스크를 검색하여 활성화합니다.

HP 9000 및 HP Integrity 서버에서 부팅 디스크의 미러를 만드는 절차는 서로 다릅니다. HP Integrity 서버에서는 파티션된 부팅 디스크를 사용합니다.



**참고:** 이 예제에는 기본 스왑 논리 볼륨의 미러 사본을 만드는 절차도 들어 있습니다. 기본 스왑 미러는 특정 디스크 또는 특정 위치에 있을 필요는 없지만 연속적인 디스크 공간에 할당되어야 합니다. 기본 스왑에 대한 미러 정책에는 미러 쓰기 캐시 및 미러 일관성 복구 메커니즘이 설정되어 있지 않는 것이 좋습니다.

기본 스왑이 미러링되고 기본 스왑 장치도 덤프 영역으로 사용되고 있으면 부팅할 때 덤프가 손실되지 않도록 하기 위해 미러 쓰기 캐시 및 미러 일관성 복구가 설정되어 있지 않아야 합니다. 이러한 옵션을 재설정하려면 시스템을 유지 관리 모드로 다시 부팅하고 `lvchange` 명령을 `-M n` 및 `-c n` 옵션과 함께 사용합니다.

## HP 9000 서버에서 부팅 디스크 미러링

미러 루트 구성을 설정하려면 디스크를 루트 볼륨 그룹에 추가하고 모든 루트 논리 볼륨을 디스크로 미러링한 후 디스크를 부팅 가능하도록 설정해야 합니다. 이 예제의 경우 추가할 디스크의 경로는 `0/1/1/0.0x1.0x0`이고 디스크에 `/dev/rdisk/disk4` 및 `/dev/disk/disk4`라는 장치 특수 파일이 포함됩니다. 다음 단계를 수행하십시오.

1. 장치 파일이 올바른 위치에 있는지 확인합니다. 예를 들면 다음과 같습니다.

```
insf -e -H 0/1/1/0.0x1.0x0
```

이제 이 디스크에 대해 다음과 같은 장치 파일이 있습니다.

```
/dev/[r]disk/disk4
```

2. 다음과 같이 부팅 가능한 물리 볼륨을 만듭니다.

- ```
# pvcreate -B /dev/rdisk/disk4
```
3. 다음과 같이 기존 루트 볼륨 그룹에 물리 볼륨을 추가합니다.

```
# vgextend /dev/vg00 /dev/disk/disk4
```
 4. 다음과 같이 부팅 영역에 부팅 유틸리티를 배치합니다.

```
# mkboot /dev/rdisk/disk4
```
 5. 다음과 같이 디스크 부팅 영역에 자동 부팅 파일을 추가합니다.

```
# mkboot -a "hpux" /dev/rdisk/disk4
```



참고: 퀴럼이 손실된 경우에만 이 디스크에서 부팅하려면 대체 문자열 `hpux -lq`를 사용하여 퀴럼 검사를 비활성화할 수 있습니다. 그러나 “복구 계획” (49 페이지)의 설명대로 세 개 이상의 물리 볼륨을 사용하고 단일 장애 지점을 제거하여 루트 볼륨 그룹을 구성하여 퀴럼 손실을 최소화하는 것이 좋습니다.

6. 미래 부팅 디스크의 논리 볼륨은 원래 부팅 디스크에 구성된 순서와 같은 순서대로 확장해야 합니다. 루트 볼륨 그룹에 있는 논리 볼륨 목록과 순서를 확인합니다. 예를 들면 다음과 같습니다.

```
# pvdisplay -v /dev/disk/disk0 | grep 'current.*0000 $'
00000 현재 /dev/vg00/lvol1 00000
00038 현재 /dev/vg00/lvol2 00000
00550 현재 /dev/vg00/lvol3 00000
00583 현재 /dev/vg00/lvol4 00000
00608 현재 /dev/vg00/lvol5 00000
00611 현재 /dev/vg00/lvol6 00000
00923 현재 /dev/vg00/lvol7 00000
01252 현재 /dev/vg00/lvol8 00000
```

7. `vg00`(루트 볼륨 그룹)의 각 논리 볼륨을 지정된 물리 볼륨에 미리링합니다. 예를 들면 다음과 같습니다.

```
# lvextend -m 1 /dev/vg00/lvol1 /dev/disk/disk4
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol2 /dev/disk/disk4
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol3 /dev/disk/disk4
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol4 /dev/disk/disk4
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol5 /dev/disk/disk4
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol6 /dev/disk/disk4
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
```

```
# lvextend -m 1 /dev/vg00/lvol17 /dev/disk/disk4
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol18 /dev/disk/disk4
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
```



팁: 미러 사본을 동기화하는 데 필요한 시간을 줄이려면 HP-UX 11i v3의 2007년 9월 릴리즈에 도입된 `lvextend` 및 `lvsync` 명령 옵션을 사용합니다. 이러한 옵션을 사용하면 순차가 아닌 병렬로 논리 볼륨을 다시 동기화할 수 있습니다. 예를 들면 다음과 같습니다.

```
# lvextend -s -m 1 /dev/vg00/lvol11 /dev/disk/disk4
# lvextend -s -m 1 /dev/vg00/lvol12 /dev/disk/disk4
# lvextend -s -m 1 /dev/vg00/lvol13 /dev/disk/disk4
# lvextend -s -m 1 /dev/vg00/lvol14 /dev/disk/disk4
# lvextend -s -m 1 /dev/vg00/lvol15 /dev/disk/disk4
# lvextend -s -m 1 /dev/vg00/lvol16 /dev/disk/disk4
# lvextend -s -m 1 /dev/vg00/lvol17 /dev/disk/disk4
# lvextend -s -m 1 /dev/vg00/lvol18 /dev/disk/disk4
# lvsync -T /dev/vg00/lvol*
```

8. 다음과 같이 루트 볼륨 그룹 정보를 업데이트합니다.


```
# lvlnboot -R /dev/vg00
```
9. 다음과 같이 미러 디스크가 부팅 디스크로 표시되는지 및 부팅, 루트 및 스왑 논리 볼륨이 두 디스크에 모두 표시되는지 확인합니다.


```
# lvlnboot -v
```
10. 다음과 같이 미러 디스크를 비휘발성 메모리의 대체 부팅 경로로 지정합니다.


```
# setboot -a 0/1/1/0.0x1.0x0
```
11. 다음과 같이 `vi`나 다른 텍스트 편집기를 사용하여 새 부팅 디스크에 대한 줄을 `/stand/bootconf`에 추가합니다.


```
# vi /stand/bootconf
1 /dev/disk/disk4
```

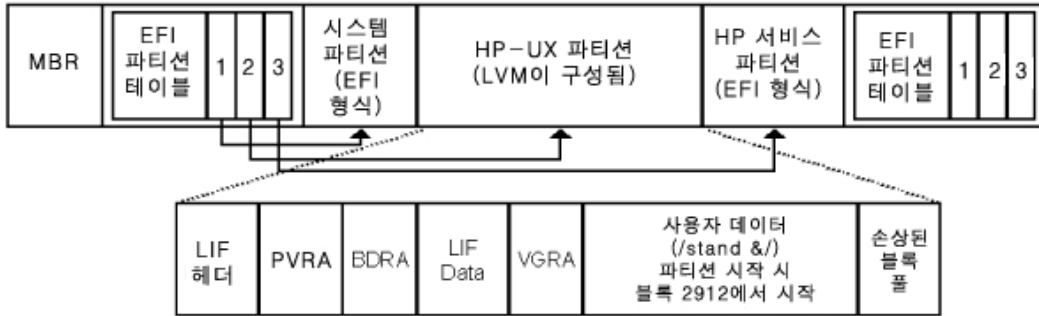
여기서 문자 “1”(소문자 L)은 LVM을 나타냅니다.

HP Integrity 서버에서 부팅 디스크 미러링

Integrity 서버에서 루트 디스크를 미러링하는 절차는 HP 9000 서버의 절차와 비슷합니다. 차이점은 Integrity 서버 부팅 디스크가 파티션된다는 점입니다. 파티션을 설정하고 유틸리티를 EFI 파티션에 복사한 후 LVM 명령에 HP-UX 파티션 장치 파일을 사용해야 합니다.

그림 3-1은 부팅 디스크의 디스크 레이아웃을 보여 줍니다. 디스크에는 MBR(Master Boot Record) 및 각 파티션을 가리키는 EFI 파티션 테이블이 들어 있습니다. idisk 명령은 파티션을 만듭니다(idisk(1M) 참조).

그림 3-1 HP Integrity 서버의 LVM 디스크 레이아웃 예제



이 예제의 경우 추가할 디스크의 하드웨어 경로는 0/1/1/0.0x1.0x0이고 /dev/disk/disk2 및 /dev/rdisk/disk2라는 장치 특수 파일이 포함됩니다. 다음 단계를 수행하십시오.

1. idisk 명령 및 파티션 설명 파일을 사용하여 디스크를 파티션합니다.

- a. 파티션 설명 파일을 만듭니다. 예를 들면 다음과 같습니다.

```
# vi /tmp/idf
```

이 예제에서 파티션 설명 파일에는 다음 정보가 포함됩니다.

```
3
EFI 500MB
HPUX 100%
HPSP 400MB
```

예제의 값은 EFI 파티션, HP-UX 파티션 및 HPSP 등의 세 개의 파티션이 있는 부팅 디스크를 표시합니다. 이전 HP Integrity 서버의 부팅 디스크에는 100MB의 EFI 파티션만이 있으며 HPSP 파티션을 포함하지 않습니다.

- b. 다음과 같이 idisk 및 파티션 설명 파일을 사용하여 디스크를 파티션합니다.

```
# idisk -f /tmp/idf -w /dev/rdisk/disk2
```

- c. 파티션이 올바르게 배치되어 있는지 확인하려면 다음 명령을 입력합니다.

```
# idisk /dev/rdisk/disk2
```

2. 모든 파티션에 대해 장치 파일을 만듭니다. 예를 들면 다음과 같습니다.

```
# insf -e -H 0/1/1/0.0x1.0x0
```

이제 이 디스크에 대해 다음과 같은 장치 파일이 있습니다.

```
/dev/[r]disk/disk2 (전체 디스크 참조)
/dev/[r]disk/disk2_p1 (efi 파티션 참조)
/dev/[r]disk/disk2_p2 (hp-ux 파티션 참조)
/dev/[r]disk/disk2_p3 (service 파티션 참조)
```

3. HP-UX 파티션을 나타내는 장치 파일을 사용하여 부팅 가능한 물리 볼륨을 만듭니다. 예를 들면 다음과 같습니다.

```
# pvcreate -B /dev/rdisk/disk2_p2
```

4. 다음과 같이 기존 루트 볼륨 그룹에 물리 볼륨을 추가합니다.

```
# vgextend vg00 /dev/disk/disk2_p2
```

5. 부팅 영역에 부팅 유틸리티를 배치합니다. EFI 파티션에 EFI 유틸리티를 복사하고 다음과 같이 전체 디스크에 대해 장치 특수 파일을 사용합니다.

```
# mkboot -e -l /dev/rdisk/disk2
```

6. 다음과 같이 디스크 부팅 영역에 자동 부팅 파일을 추가합니다.

```
# mkboot -a "hpx" /dev/rdisk/disk2
```



참고: 쿼럼이 손실된 경우에만 이 디스크에서 부팅하려면 대체 문자열 `hpx -lq`를 사용하여 쿼럼 검사를 비활성화할 수 있습니다. 그러나 “복구 계획” (49 페이지)의 설명대로 세 개 이상의 물리 볼륨을 사용하고 단일 장애 지점을 제거하여 루트 볼륨 그룹을 구성하여 쿼럼 손실을 최소화하는 것이 좋습니다.

7. 미리 부팅 디스크의 논리 볼륨은 원래 부팅 디스크에 구성된 순서와 같은 순서대로 확장해야 합니다. 루트 볼륨 그룹에 있는 논리 볼륨 목록과 순서를 확인합니다. 예를 들면 다음과 같습니다.

```
# pvdisplay -v /dev/disk/disk0_p2 | grep 'current.*0000 $'
00000 현재 /dev/vg00/lvol1 00000
00010 현재 /dev/vg00/lvol2 00000
00138 현재 /dev/vg00/lvol3 00000
00151 현재 /dev/vg00/lvol4 00000
00158 현재 /dev/vg00/lvol5 00000
00159 현재 /dev/vg00/lvol6 00000
00271 현재 /dev/vg00/lvol7 00000
00408 현재 /dev/vg00/lvol8 00000
```

8. `vg00`(루트 볼륨 그룹)의 각 논리 볼륨을 지정된 물리 볼륨에 미리링합니다. 예를 들면 다음과 같습니다.

```
# lvextend -m 1 /dev/vg00/lvol1 /dev/disk/disk2_p2
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol2 /dev/disk/disk2_p2
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol3 /dev/disk/disk2_p2
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol4 /dev/disk/disk2_p2
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol5 /dev/disk/disk2_p2
이제 새로 할당된 미러가 동기화됩니다.
```

```

이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol6 /dev/disk/disk2_p2
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol7 /dev/disk/disk2_p2
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....
# lvextend -m 1 /dev/vg00/lvol8 /dev/disk/disk2_p2
이제 새로 할당된 미러가 동기화됩니다.
이 작업은 다소 시간이 걸립니다. 잠시 기다려주세요 ....

```



참고: lvextend가 실패하고 다음 메시지가 표시되는 경우:

"m": 잘못된 옵션입니다.

HP MirrorDisk/UX가 설치되지 않습니다.



팁: 미러 사본을 동기화하는 데 필요한 시간을 줄이려면 HP-UX 11i v3의 2007년 9월 릴리즈에 도입된 lvextend 및 lvsync 명령 옵션을 사용합니다. 이러한 옵션을 사용하면 순차가 아닌 병렬로 논리 볼륨을 다시 동기화할 수 있습니다. 예를 들면 다음과 같습니다.

```

# lvextend -s -m 1 /dev/vg00/lvol1 /dev/disk/disk2_p2
# lvextend -s -m 1 /dev/vg00/lvol2 /dev/disk/disk2_p2
# lvextend -s -m 1 /dev/vg00/lvol3 /dev/disk/disk2_p2
# lvextend -s -m 1 /dev/vg00/lvol4 /dev/disk/disk2_p2
# lvextend -s -m 1 /dev/vg00/lvol5 /dev/disk/disk2_p2
# lvextend -s -m 1 /dev/vg00/lvol6 /dev/disk/disk2_p2
# lvextend -s -m 1 /dev/vg00/lvol7 /dev/disk/disk2_p2
# lvextend -s -m 1 /dev/vg00/lvol8 /dev/disk/disk2_p2
# lvsync -T /dev/vg00/lvol*

```

9. 다음과 같이 루트 볼륨 그룹 정보를 업데이트합니다.

```
# lvlnboot -R /dev/vg00
```

10. 다음과 같이 미러 디스크가 부팅 디스크로 표시되는지 및 부팅, 루트 및 스왑 논리 볼륨이 두 디스크에 모두 표시되는지 확인합니다.

```
# lvlnboot -v
```

11. 다음과 같이 미러 디스크를 비휘발성 메모리의 대체 부팅 경로로 지정합니다.

```
# setboot -a 0/1/1/0.0x1.0x0
```

12. 다음과 같이 vi나 다른 텍스트 편집기를 사용하여 새 부팅 디스크에 대한 줄을 /stand/bootconf에 추가합니다.

```

# vi /stand/bootconf
1 /dev/disk/disk2_p2

```


여기서 문자 “1”(소문자 l)은 LVM을 나타냅니다.

파일 시스템 논리 볼륨 관리

이 절에서는 논리 볼륨 내부의 파일 시스템을 통해 사용할 경우 수행해야 하는 특별한 작업에 대해 설명합니다. 다음 내용에 대해 설명합니다.

- “파일 시스템 만들기” (105 페이지)
- “파일 시스템 확장” (106 페이지)
- “파일 시스템 크기 축소” (108 페이지)
- “VxFS 스냅샷 파일 시스템 백업” (110 페이지)



팁: 파일 시스템을 처리할 때 HP SMH 또는 일련의 HP-UX 명령을 사용할 수 있습니다. 대부분의 작업에서는 HP SMH를 사용하는 것이 더 빠르고 간단합니다. 다음 작업을 따로 수행할 필요가 없습니다. 그보다 HP SMH 디스크 및 파일 시스템 영역에서 작업을 진행하는 것이 좋습니다. 그러면 HP SMH에서 필요한 단계를 모두 수행하게 됩니다.

파일 시스템 만들기

논리 볼륨에서 HFS 또는 VxFS 파일 시스템을 만들 경우 HP SMH 또는 일련의 HP-UX 명령을 사용할 수 있습니다. HP-UX 명령을 직접 사용하도록 선택한 경우 다음은 파일 시스템을 만드는 하위 작업에 대해 설명하는 목록입니다.

1. HFS가 아닌 종류의 새 파일 시스템을 만드는 경우 새 종류를 커널에 재구성해야 할 수 있습니다. 일반적으로 VxFS는 기본 구성의 일부로 커널에 구성되어 있습니다. 파일 시스템 유형 추가 방법에 대한 자세한 내용은 **HP-UX 시스템 관리 설명서: 구성 관리**를 참조하십시오.
2. 논리 볼륨에 필요한 크기를 추정합니다. 파일 시스템이 들어 있는 논리 볼륨에 필요한 크기를 추정하려면 “파일 시스템용 논리 볼륨 설정” (34 페이지)을 참조하십시오.
3. 볼륨 그룹에 사용 가능한 공간이 충분한지 확인합니다. “논리 볼륨 만들기” (63 페이지)의 설명에 따라 `vgdisplay` 명령을 사용하여 이 정보를 계산합니다.
볼륨 그룹에 공간이 부족하면 “볼륨 그룹에 디스크 추가” (62 페이지)의 설명에 따라 볼륨 그룹에 디스크를 추가할 수 있습니다.
4. 논리 볼륨을 만듭니다. “논리 볼륨 만들기” (63 페이지)의 설명에 따라 `lvcreate`를 사용합니다.

5. 특성 장치 파일을 사용하여 파일 시스템을 만듭니다. 예를 들면 다음과 같습니다.

```
# newfs -F fstype /dev/vg02/rlvol1
```

-F fstype 옵션을 사용하지 않으면 newfs는 /etc/fstab 파일의 내용에 따라 파일 시스템을 만듭니다. /etc/fstab에 파일 시스템에 대한 항목이 없으면 /etc/default/fs 파일에서 파일 시스템 종류가 결정됩니다. 추가 옵션에 대한 자세한 내용은 newfs(1M)를 참조하십시오.

VxFS 파일 시스템을 만들 때 파일 이름은 자동으로 긴 이름이 됩니다.

HFS의 경우 -s 또는 -l 옵션을 사용하여 파일 시스템에 파일 이름을 각각 짧게 또는 길게 지정합니다. 기본적으로 파일 시스템 이름의 길이는 루트 파일 시스템의 이름 길이와 일치합니다. 짧은 파일 이름은 최대 14자까지, 긴 파일 이름은 255자까지 사용할 수 있습니다. 용통성을 위해 긴 파일 이름을 사용하는 것이 좋습니다. 이렇게 하면 긴 파일 이름을 사용하는 다른 시스템에서 만든 파일의 이름을 다시 지정하지 않고도 해당 시스템으로 파일을 이동할 수 있습니다.

6. 파일 시스템을 만든 후 사용자가 액세스할 수 있도록 마운트하고 부팅 시 자동으로 마운트되도록 /etc/fstab에 추가합니다.

파일 시스템 확장

논리 볼륨 내부에서 파일 시스템 확장은 2단계의 작업으로 논리 볼륨을 확장한 다음 파일 시스템을 확장하는 것입니다. 첫 단계는 “논리 볼륨 확장” (65 페이지)에서 설명합니다. 파일 시스템 자체를 확장하는 두 번째 단계는 다음과 같은 요소에 따라 달라집니다.

- 관련된 파일 시스템 종류는 무엇입니까? HFS입니까 아니면 VxFS입니까? HFS의 경우 확장하려면 파일 시스템을 마운트 해제해야 합니다.

파일 시스템 종류를 확인합니다. 예를 들면 다음과 같습니다.

```
# /usr/sbin/fstyp /dev/vg01/lvol2
vxfs
```

- 파일 시스템이 VxFS인 경우 기본 VxFS 제품이 있습니까 아니면 OnlineJFS 제품이 있습니까? 기본 VxFS 제품만 있는 경우 확장하기 전에 파일 시스템을 마운트 해제해야 합니다.

OnlineJFS 제품이 설치되어 있는지 확인하려면 다음 명령을 입력합니다.

```
# swlist -l product | grep -i OnlineJFS
OnlineJFS B.11.31 Online features of the VxFS File System
```

- 파일 시스템을 마운트 해제할 수 있습니까? /var 및 /usr과 같은 시스템 디렉토리를 마운트 해제하려면 단일 사용자 모드 상태여야 합니다.
- 파일 시스템이 루트 파일 시스템(/)입니까? 루트 파일 시스템인 경우 다음 두 가지 복잡한 문제가 있습니다.
 - 루트 파일 시스템이 들어 있는 논리 볼륨은 연속 할당 정책을 통해 만들어지므로 제자리에서 확장할 수 없습니다.
 - 단일 사용자 상태로 변경하는 경우에도 루트 파일 시스템을 마운트 해제할 수 없습니다.

VxFS를 루트 파일 시스템으로 사용하고 OnLineJFS 제품이 있는 경우 사용 가능한 연속적인 디스크 공간이 있으면 마운트 해제하지 않고 원래 루트 파일 시스템을 확장할 수 있습니다.

그렇지 않은 경우 현재 루트 파일 시스템을 확장하려면 마운트 해제된 원래 루트 디스크를 사용할 수 있도록 하는 **다른** 루트 디스크를 만들고 마운트해야 하므로 사용 가능한 연속적인 디스크 공간이 계속 있는 **경우** 현재 루트 파일 시스템이 확장됩니다. 원래 디스크에 사용 가능한 연속적인 디스크 공간이 없으면 원래 루트 디스크를 확장하는 대신 용량이 더 큰 다른 디스크에 새 루트 파일 시스템을 만들 수 있습니다.



주의: 이 절차는 이미 100% 용량에 도달한 VxFS 파일 시스템에서 실패할 수 있습니다 (오류 28). 이 작업을 시도하기 전에 일부 파일을 제거해야 합니다.

이러한 질문에 대답한 후 다음 단계를 수행합니다.

1. 마운트 해제해야 하는 파일 시스템을 마운트 해제합니다.
 - a. 이 논리 볼륨에 마운트된 파일 시스템에 열린 파일이 없고 사용자에게 현재 작업 중인 디렉토리가 없는지 확인합니다. 예를 들면 다음과 같습니다.

```
# fuser -cu /work/project5
```

논리 볼륨이 사용 중이면 기본 응용 프로그램에 해당 논리 볼륨이 더 이상 필요하지 않은지 확인합니다. 필요한 경우 응용 프로그램을 중지합니다.



참고: NFS를 사용하여 다른 시스템으로 파일 시스템을 내보낼 경우 이 시스템을 사용하는 사용자가 없는지 확인한 다음 해당 시스템에서 파일 시스템을 마운트 해제합니다.

- b. 논리 볼륨을 사용하는 응용 프로그램을 중지할 수 없거나 /var나 /usr과 같은 시스템 디렉토리인 경우 다음과 같이 단일 사용자 상태로 변경합니다.

```
# /sbin/shutdown
```

- c. 다음과 같이 파일 시스템을 마운트 해제합니다.

```
# /sbin/umount /dev/vg01/lvol2
```

2. 논리 볼륨을 확장합니다. 예를 들면 다음과 같습니다.

```
# /sbin/lvextend -L 332 /dev/vg01/lvol2
```

명령은 이 볼륨의 크기를 332MB로 증가시킵니다.

3. 파일 시스템 크기를 논리 볼륨 크기로 확장합니다. 파일 시스템이 마운트 해제되는 경우 다음과 같이 `extendfs` 명령을 사용합니다.

```
# /sbin/extendfs /dev/vg01/rlvol2
```

파일 시스템을 마운트 해제할 필요가 없다면 `fsadm` 명령을 대신 사용합니다. 파일 시스템의 블록 크기를 기준으로 새 크기가 지정됩니다. 이 예제에서 파일 시스템 /work/project5의 블록 크기는 1KB입니다. 파일 시스템을 332MB로 확장하려면 블록 수는 339968(332 x 1024)입니다. 예를 들면 다음과 같습니다.

```
# fsadm -b 339968 /work/project5
```

4. 파일 시스템을 마운트 해제한 경우 다시 마운트합니다.

- a. 단일 사용자 상태로 변경한 경우 시스템을 다시 부팅합니다.

```
# /sbin/reboot -r
```

부팅 과정에서 파일 시스템을 마운트하고 내보내므로 파일 시스템을 마운트하고 내보내는 추가 단계를 건너뛸 수 있습니다.

- b. 다음과 같이 파일 시스템을 다시 마운트합니다.

```
# /sbin/mount /dev/vg01/rlvol2 /mount_point
```



참고: NFS 클라이언트에서 파일 시스템을 계속 사용할 예정이면 서버에 내보내고(`exportfs -a`) 클라이언트에 다시 마운트합니다(`mount -a`).

5. `bdf`, `df` 또는 `fsadm -E`를 입력하여 파일 시스템이 확장을 반영하는지 확인합니다.

파일 시스템 크기 축소

필요량보다 많은 디스크 공간이 할당된 파일 시스템을 줄여서 일부 다른 용도를 위해 디스크 공간을 해제할 수 있습니다.

파일 시스템 크기를 축소하는 작업은 확장하는 작업보다 더 복잡합니다. 파일 시스템 블록 할당 전략으로 인해 데이터는 논리 볼륨 전체에 흩어져 있을 수 있습니다. 논리 볼륨을 줄이면 논리 볼륨 끝의 공간을 되찾으므로 파일 시스템 드라이버가 미리 데이터 블록을 결합 및 재정렬해야 합니다. 대부분의 파일 시스템 종류는 이러한 결합을 수행할 수 없으므로 파일 시스템의 데이터를 백업하고 논리 볼륨을 줄인 후 더 작은 논리 볼륨에 새로운 파일 시스템을 만들고 백업에서 데이터를 복원해야 합니다.

온라인 통합 및 크기 축소를 수행할 수 있는 현재 파일 시스템 종류는 OnlineJFS뿐이고 경우에 따라 실패할 수 있습니다.

OnlineJFS를 통해 만들어진 파일 시스템 축소

`fsadm` 명령을 사용하면 이 명령이 할당 취소를 시도하는 블록이 현재 사용 중이 아닌 경우 파일 시스템이 축소되고 사용 중인 경우 명령이 실패합니다. 현재 충분한 여유 공간을 사용할 수 없는 경우 디렉토리 및 확장 영역의 파일 시스템 조각 모음을 통해 사용 가능한 공간을 파일 시스템 끝쪽으로 통합하여 다시 시도할 때 축소 프로세스가 성공하도록 할 수 있습니다.

예를 들어, VxFS 파일 시스템이 현재 6GB라고 가정합니다. 그러나 실제로는 2GB와 예약 공간으로 추가 1GB만 필요합니다. 따라서 파일 시스템 크기를 3GB로 조정합니다. `fsadm`에 `-b` 옵션을 사용하여 섹터에서 새 파일 시스템 크기를 지정한 다음 일치하도록 논리 볼륨 크기를 축소시킵니다. 파일 시스템 섹터 크기가 1K라고 가정하는 경우 다음 명령을 사용합니다.

```
# fsadm -b 3145728 /home
# lvreduce -L 3072 /dev/vg01/lvol5
```

HFS 또는 VxFS로 만들어진 파일 시스템 축소

HFS 또는 VxFS로 만든 파일 시스템의 크기를 축소하려면 다음 단계를 수행합니다.

1. 논리 볼륨의 파일 시스템에 열린 파일이 없고 논리 볼륨이 현재 작업 중인 디렉토리가 아닌지 확인합니다. 예를 들면 다음과 같습니다.

```
# fuser -cu /dev/vg01/lvol5
```



참고: NFS를 사용하여 다른 시스템으로 파일 시스템을 내보낼 경우 이 시스템을 사용하는 사용자가 없는지 확인한 다음 서버에서 마운트 해제하기 전에 해당 시스템에서 파일 시스템을 마운트 해제합니다.

2. 논리 볼륨의 데이터를 백업합니다.

예를 들어 /work/project5를 시스템 기본 테이프 장치에 백업하려면 다음 명령을 입력합니다.

```
# tar cv /work/project5
```

3. 다음과 같이 논리 볼륨이 마운트된 파일 시스템의 데이터를 제거합니다.

```
# rm -r /work/project5
```

/work/project5는 마운트 지점이므로 rm -r로 디렉토리 자체가 제거되지 않습니다.

4. 다음과 같이 논리 볼륨이 마운트된 파일 시스템을 마운트 해제합니다.

```
# umount /work/project5
```

5. 다음과 같이 논리 볼륨의 크기를 축소합니다.

```
# lvreduce -L 500 /dev/vg01/lvol5
```

이 명령은 논리 볼륨 /dev/vg01/lvol5를 500MB로 축소시킵니다.

6. 특정 장치 파일을 사용하여 축소된 논리 볼륨에 새 파일 시스템을 만듭니다. 예를 들면 다음과 같습니다.

```
# newfs -f fstype /dev/vg01/rlvol5
```

7. 다음과 같이 논리 볼륨을 마운트합니다.

```
# mount /dev/vg01/lvol5 /work/project5
```

8. 백업에서 데이터를 복구합니다. 예를 들면 다음과 같습니다.

```
# tar xv
```

명령을 실행하면 테이프의 모든 내용이 시스템 기본 드라이브에 복구됩니다.

9. NFS 클라이언트에서 /work/project5를 계속 사용하려면 서버에 다시 내보내고 (exportfs -a) 클라이언트에 다시 마운트합니다(mount -a).



참고: VxFS 스냅샷 파일 시스템을 만들고 백업하려면 선택적인 HP OnlineJFS 제품이 시스템에 설치되어 있어야 합니다. 자세한 내용은 **HP-UX 시스템 관리 설명서: 구성 관리**를 참조하십시오.

VxFS를 사용하면 그 순간에 파일 시스템의 읽기 전용 이미지인 파일 시스템의 스냅샷을 만들어 파일 시스템을 오프라인으로 만들지 않고도 백업을 수행할 수 있습니다. 기본 파일 시스템은 온라인 상태로 유지되며 계속 변경됩니다. 스냅샷을 만든 후에는 `dump` 명령을 제외하고 모든 백업 유틸리티를 사용하여 백업할 수 있습니다.

VxFS 스냅샷 파일 시스템을 만들어 백업하려면 다음 단계를 수행합니다.

1. 스냅샷 파일 시스템의 크기를 결정한 다음 이를 포함할 논리 볼륨을 만듭니다. `bdf`를 사용하여 기본 파일 시스템 크기를 평가하고 다음을 고려하십시오.
 - 파일 시스템의 블록 크기(기본적으로 블록당 1,024바이트)
 - 이 파일 시스템에서 변경될 것으로 예상되는 데이터의 양(총 파일 시스템 크기의 15~20%를 사용하는 것이 좋음)

예를 들어, `/home` 디렉토리에 마운트된 `lv014`의 스냅샷을 만들 크기를 결정하려면 `bdf` 출력을 확인합니다.

```
# bdf /home
filesystem      kbytes    used    avail %used mounted on
/dev/vg00/lvol4    40960    38121    2400   94% /home
```

이 파일 시스템(40MB)의 20%가 변경될 수 있다고 생각하면 블록 8개에 해당하는 논리 볼륨(8MB)을 만듭니다.

2. `lvcreate`를 사용하여 스냅샷 파일 시스템이 들어가도록 논리 볼륨을 만듭니다. 예를 들면 다음과 같습니다.

```
# lvcreate -L 8 -n snap /dev/vg02
```

이 명령은 `/dev/vg02/snap`이라는 8MB의 논리 볼륨을 만듭니다. 이 볼륨에는 `lv014`의 스냅샷 파일 시스템이 포함될 수 있습니다.

3. 스냅샷 파일 시스템의 마운트 지점에 대한 디렉토리를 만듭니다. 예를 들면 다음과 같습니다.

```
# mkdir /tmp/house
```

4. 스냅샷 파일 시스템을 만들고 마운트합니다. 예를 들면 다음과 같습니다.

```
# mount -f vxfs -o snapof=/dev/vg00/lvol4 /dev/vg02/snap /tmp/house
```

이 예제에서 스냅샷은 논리 볼륨 `/dev/vg00/lvol4`에서 가져와 논리 볼륨 `/dev/vg02/snap`에 포함된 다음 `/tmp/house`에 마운트합니다.

5. `dump`를 제외한 모든 백업 유틸리티를 사용하여 스냅샷 파일 시스템을 백업합니다. 예를 들어 다음과 같이 `tar`을 사용하여 스냅샷 파일 시스템 `/tmp/house`를 보관하면 테이프에 있는 파일이 상대 경로 이름을 갖게 됩니다.

```
# cd tmp; tar cf /dev/rtape/tape0BEST house
```

또는 다음 `vxdump` 명령을 사용하면 확장 영역 속성을 갖는 스냅샷 파일 시스템 `/tmp/house`가 백업됩니다.

```
# vxdump -0 -f /dev/rtape/tape0BEST /tmp/house
```

스왑 논리 볼륨 관리



참고: 버전 2.x 볼륨 그룹은 스왑 논리 볼륨을 지원하지 않습니다.

논리 볼륨에 스왑 영역을 사용하게 되면 HP-UX가 영역의 크기를 판단하고 그 이상의 공간은 사용하지 않게 됩니다. 디스크에 연속적인 공간이 충분히 남아 있으면 `lvextend` 명령(또는 HP SMH)을 사용하여 논리 볼륨의 크기를 늘린 다음 시스템을 다시 부팅하여 기본 스왑 영역의 크기를 늘릴 수 있습니다. 이 절차를 통해 HP-UX는 추가 공간을 사용할 수 있습니다.

기본 스왑에 추가로 장치 스왑 영역을 사용할 경우 장치 스왑 영역이 여러 물리 볼륨에 있을 때 성능이 가장 좋게 됩니다. 이렇게 구성하면 스와핑이 발생할 때 물리 볼륨에 I/O를 인터리브할 수 있게 됩니다.

인터리브 스왑을 만들려면 개별 디스크에 각각 논리 볼륨을 만들어 스왑에 대해 여러 논리 볼륨을 만듭니다. 이 구성을 얻으려면 HP-UX 명령을 사용해야 합니다. HP SMH에서는 특정 디스크에 논리 볼륨을 만들 수 없습니다. 자세한 내용은 “논리 볼륨을 특정 디스크로 확장” (66 페이지)을 참조하십시오.

HP-UX 시스템 관리 설명서: 개요의 설명대로 스왑 공간을 구성할 수 있습니다.



참고: 시스템을 다시 부팅해야 시스템에서 스왑 구성의 변경 사항을 인식합니다.

스왑 논리 볼륨 만들기

스왑 논리 볼륨을 만들려면 `lvcreate` 명령을 사용합니다. `-C y` 옵션을 사용하여 연속 할당 정책을 설정해야 합니다. 예를 들면 다음과 같습니다.

```
# lvcreate -C y -n swap_lvol /dev/vgmn
```

자세한 내용은 `lvcreate(1M)`를 참조하십시오.

기본 스왑으로 사용할 논리 볼륨을 만들고 나면 `lvlnboot` 명령에 `-s` 옵션을 사용하여 부팅 시에 LVM에서 사용하는 스왑 정보를 업데이트합니다. 예를 들면 다음과 같습니다.

```
# lvlnboot -s /dev/vgmn/swap_lvol
```

스왑 장치 확장

스왑에 논리 볼륨을 사용 중이라면 스왑 크기를 늘리기 전에 논리 볼륨 크기를 늘려야 합니다. `lvextend` 또는 HP SMH를 사용하여 논리 볼륨을 확장할 수 있습니다.

스왑 논리 볼륨이 연속적이어야 하기 때문에 기존 논리 볼륨의 끝에 사용 가능한 물리 확장 영역이 있는 경우에만 논리 볼륨 확장이 성공합니다. 연속적인 디스크 공간을 사용

할 수 없으면 루트 볼륨 그룹 안에 기본 스왑에 대한 새 연속 논리 볼륨을 만들어야 합니다. 이 경우에는 특정 디스크를 지정할 필요가 없습니다. 예를 들면 다음과 같습니다.

```
# lvcreate -C y -L 48 -n pswap /dev/vgroot
```

기본 스왑으로 사용할 논리 볼륨을 만든 후 `lvlnboot`를 사용하여 부팅 정보를 업데이트합니다.

```
# lvlnboot -s /dev/vgroot/pswap
```

스왑 장치 크기 축소

스왑에 논리 볼륨을 사용 중이라면 논리 볼륨의 크기를 축소시키기 전에 스왑 크기를 줄여야 합니다. `lvreduce` 또는 `HP SMH`를 사용하여 논리 볼륨의 크기를 축소시킬 수 있습니다.

덤프 논리 볼륨 관리



참고: 버전 2.x 볼륨 그룹은 덤프 논리 볼륨을 지원하지 않습니다.

이 절에서는 덤프 장치로 논리 볼륨 사용에 대한 LVM 정보를 설명합니다. 덤프 장치 구성 및 관리에 대한 자세한 내용은 **HP-UX 시스템 관리 설명서: 개요**를 참조하십시오.

덤프 논리 볼륨 만들기

덤프 논리 볼륨을 만들려면 `lvcreate` 명령을 사용합니다. `-C y` 옵션을 사용하여 연속 할당 정책을 설정하고 `-r n`을 사용하여 손상된 블록 배치를 비활성화해야 합니다. 예를 들면 다음과 같습니다.

```
# lvcreate -C y -r n -n dump_lvol /dev/vgmn
```

자세한 내용은 `lvcreate(1M)`를 참조하십시오.

덤프 장치로 사용할 논리 볼륨을 만들고 나면 `lvlnboot` 명령에 `-d` 옵션을 사용하여 LVM에서 사용하는 덤프 정보를 업데이트합니다. 덤프 영역으로 사용할 논리 볼륨 `/dev/vg00/lvol2`를 만든 경우 다음을 입력하여 부팅 정보를 업데이트합니다.

```
# lvlnboot -d /dev/vg00/lvol2
```

덤프 논리 볼륨 제거

현재 구성된 논리 볼륨을 덤프 장치로 사용하지 않으려면 `lvrmboot` 명령에 `-d` 옵션을 사용합니다. 예를 들면 다음과 같습니다.

```
# lvrmboot -d /dev/vg00/lvol2
```

그런 다음 다른 용도로 논리 볼륨을 사용할 수 있습니다. 논리 볼륨을 완전히 제거하려면 “논리 볼륨 제거” (69 페이지)를 참조하십시오.

하드웨어 문제

이 절에서는 LVM을 처리하는 하드웨어 관련 문제에 대해 설명합니다.

복제된 LUN 통합

특정 디스크 어레이는 해당 LUN의 복제본을 만들 수 있습니다. 예를 들어, HP XP 제품을 사용하여 기존 LUN의 사본인 BC(Business Copies)라는 LUN 집합을 분리할 수 있습니다.

복제된 디스크의 LVM 헤더에는 원래 디스크와 동일한 정보가 있습니다. 이는 각 디스크에 고유 식별자가 있다는 LVM의 요구 사항을 위반합니다. 복제된 디스크를 LVM과 함께 사용할 수 있도록 하려면 `vgchgid` 명령을 사용하여 VGID(볼륨 그룹 식별자)를 변경합니다.

변경할 모든 물리 볼륨은 동일한 볼륨 그룹에 속해야 합니다. 따라서 여러 물리 볼륨을 변경할 경우 `vgchgid`를 한 번 호출하여 모든 물리 볼륨을 지정합니다. 그렇지 않으면 물리 볼륨에 서로 다른 VGID가 할당됩니다.

예를 들어, 4개의 물리 볼륨이 들어 있는 볼륨 그룹이 있고 각 물리 볼륨에 대해 BC를 만든다고 가정합니다. BC 두 개에 대해서만 `vgchgid`를 실행하면 `vgchgid`는 해당 두 개의 BC에 대해서 VGID를 수정합니다. 그런 다음 네 개의 BC 모두에 대해 다시 `vgchgid`를 실행하면 `vgchgid`는 BC가 서로 다른 볼륨 그룹에 속한다고 보고합니다. 이 문제를 해결하려면 수정된 두 BC에 대해 `vgchgid`를 실행한 다음 두 개의 개별 볼륨에서 네 개의 BC를 사용하거나 수정된 두 개의 BC를 병합하고 마지막으로 모든 네 개의 BC에 대해 `vgchgid`를 실행하기 전에 다시 분리할 수 있습니다.

물리 볼륨 집합에 대해 `vgchgid`를 실행한 후 `vgimport`를 사용하여 새 볼륨 그룹으로 가져옵니다. 예를 들면 다음과 같습니다.

1. BC 사본을 만들고 어레이에 대한 명령을 사용하여 새 장치 파일을 만듭니다.
2. 다음과 같이 복제된 디스크의 VGID를 변경합니다.

```
# vgchgid /dev/rdisk/disk49 /dev/rdisk/disk50
```
3. 2008년 3월 이전의 HP-UX 릴리즈를 사용하고 있는 경우 “볼륨 그룹 장치 파일 만들기” (60 페이지)의 절차를 사용하여 볼륨 그룹 group 파일을 만듭니다.
4. 다음과 같이 물리 볼륨을 가져옵니다.

```
# vgimport /dev/vg04 /dev/rdisk/disk49 /dev/rdisk/disk50
```
5. 다음과 같이 볼륨 그룹 구성 정보를 백업합니다.

```
# vgcfgbackup /dev/vg04
```
6. 다음과 같이 볼륨 그룹을 활성화합니다.

```
# vgchange -a y /dev/vg04
```


4 LVM 문제 해결

이 장에서는 다음과 같은 내용을 다룹니다.

- “문제 해결 도구 개요” (115 페이지)
- “로그 파일 및 추적 파일” (117 페이지)
- “I/O 오류” (117 페이지)
- “볼륨 그룹 활성화 장애” (120 페이지)
- “LVM 부팅 장애” (124 페이지)
- “논리 볼륨의 크기를 축소시킨 후 발생하는 문제” (125 페이지)
- “손상된 디스크 교체” (126 페이지)
- “경고 및 오류 메시지” (138 페이지)
- “문제 보고” (151 페이지)

문제 해결 도구 개요

이 절에서는 LVM 문제 해결에 사용할 수 있는 도구에 대해 설명합니다.

정보 모음

`vgdisplay`, `lvdisplay`, `pvddisplay` 및 `lvlnboot` 명령을 사용하여 LVM 구성 정보를 수집할 수 있습니다. “복구 계획” (49 페이지)에서 설명한 대로 표 4-1에 나열된 명령에서 출력력을 정기적으로 수집합니다.

표 4-1 수집 및 유지 관리할 LVM 정보

명령	범위	목적
<code>ioscan -f</code>		I/O 구성 인쇄
<code>lvlnboot -v</code>		루트, 부팅, 스왑 및 덤프 논리 볼륨에 대한 정보 인쇄
<code>vgcfgrestore -l</code>	모든 볼륨 그룹	백업 파일에서 볼륨 그룹 구성 인쇄
<code>vgdisplay -v</code>	모든 볼륨 그룹	논리 볼륨과 물리 볼륨의 상태를 비롯한 볼륨 그룹 정보 인쇄
<code>lvdisplay -v</code>	모든 논리 볼륨	논리 확장 영역의 매핑 및 상태를 비롯한 논리 볼륨 정보 인쇄
<code>pvddisplay -v</code>	모든 물리 볼륨	물리 확장 영역의 상태를 비롯한 물리 볼륨 정보 인쇄

또한 다음 두 가지 용도로 `lvmdadm` 명령을 사용합니다.

- HP-UX 11i v3 릴리즈에서 지원하는 볼륨 그룹 버전을 확인하려는 경우. 예를 들어, 사용하는 릴리즈에서 버전 2.1 볼륨 그룹을 지원하는 경우 `lvmdadm`은 다음을 표시합니다.

```
# lvmadm -t -V 2.1
--- LVM 제한 ---
VG 버전                                2.1
최대 VG 크기(TB)                       2048
최대 LV 크기(TB)                       256
최대 PV 크기(TB)                       16
최대 VG                                2048
최대 LV 수                             2047
최대 PV 수                             2048
최대 미러 수                           5
최대 스트라이프 수                     511
최대 스트라이프 크기(KB)               262144
LV당 최대 LX 수                        33554432
PV당 최대 PX 수                        16777216
최대 확장 영역 크기(MB)                256
```

사용하는 릴리즈에서 버전 2.1 볼륨 그룹을 지원하지 않는 경우에는 다음을 표시합니다.

```
# lvmadm -t -V 2.1
오류: 2.1은(는) 잘못된 볼륨 그룹 버전입니다.
```

- 사람이 읽을 수 있는 형식으로 /etc/lvmtab 및 /etc/lvmtab_p 파일의 내용을 표시하려는 경우. 예를 들어, 다음 명령은 시스템의 모든 버전 1.0 볼륨 그룹에 대한 LVM 구성 파일의 내용을 표시합니다.

```
# lvmadm -l -V 1.0
--- 버전 1.0 볼륨 그룹 ---
VG 이름 /dev/vg00
PV 이름 /dev/disk/disk34_p2
```

일관성 검사

대부분의 LVM 명령은 일관성 검사를 수행합니다. `vgdisplay`, `lvdisplay` 및 `pvdisplay` 명령을 사용하여 LVM 구성을 검사하고 불일치를 찾을 수 있습니다.

또한 `pvck` 명령은 물리 볼륨에 대해 명시적 일관성 검사를 수행합니다. 이 명령은 백워드 시스템 마이그레이션 이후 포워드 시스템 마이그레이션으로 인한 잘못된 체크섬을 감지합니다. 따라서 비활성화된 볼륨 그룹에서만 `pvck`를 실행합니다. 자세한 내용은 `pvck(1M)`를 참조하십시오.



참고: `pvck` 명령은 버전 2.x 볼륨 그룹을 지원하지 않습니다.

유지 관리 모드 부팅

LVM **유지 관리 모드 부팅**은 일반적인 LVM 구조를 건너뛰고 시스템을 부팅하는 특별한 방법입니다. 이 방법은 시스템이 다른 방법으로 부팅되지 않도록 해야 하는 문제에만 사용해야 합니다. 정상적으로 시작되는 프로세스의 많은 부분이 시작되지 않거나 시스템 검사의 많은 부분이 수행되지 않는 단일 사용자 상태와 비슷합니다. LVM 유지 관리 모드는 일반적으로 `vgcfgrestore`를 사용하여 손상된 시스템 LVM 데이터 구조를 복구할 동안 시스템을 부팅하고 정상적으로 시스템을 부팅할 수 있도록 합니다.

일반적으로 부팅 로더는 LIF 볼륨의 LABEL 파일을 사용하여 부팅 파일 시스템의 위치 및 커널 /stand/vmunix를 확인합니다. LABEL 파일에는 루트 파일 시스템의 시작 블록과 크기도 들어 있습니다.

유지 관리 모드 부팅에서 부팅 로더는 LIF 볼륨의 정보를 사용하는 대신 부팅 디스크 사용자 데이터 영역의 시작 부분에서 부팅 파일 시스템을 찾으려고 합니다. 루트 파일 시스템의 시작 블록과 크기를 얻기 위해 부팅 로더는 /stand/rootconf 파일을 읽습니다. LVM이 활성화되어 있지 않으므로 루트 파일 시스템은 인접하게 할당되어야 합니다.

유지 관리 모드 부팅은 다음과 같은 점에서 표준 부팅과 다릅니다.

- 시스템이 단일 사용자 모드로 부팅됩니다.
- 볼륨 그룹이 활성화되지 않습니다.
- 기본 스왑 및 덤프를 사용할 수 없습니다.
- 루트 파일 시스템과 부팅 파일 시스템만 사용할 수 있습니다.
- 루트 파일 시스템이 미러링되면 하나의 사본만 사용됩니다. 루트 파일 시스템의 변경 내용은 미러 사본에 전파되지 않지만 해당 미러 사본이 못 쓰게 된 것으로 표시되므로 시스템이 정상적으로 부팅될 때 동기화됩니다.

LVM을 통해 구성된 루트 디스크가 있는 시스템에서 유지 관리 모드로 부팅하려면 부팅 로더에 -lm 옵션을 사용합니다. HP 9000 서버에서 다음 명령을 입력합니다.

```
ISL> hpux -lm
```

HP Integrity 서버에서 다음 명령을 입력합니다.

```
HPUX> boot -lm
```



주의: 유지 관리 모드로 시스템을 부팅하는 경우 루트 볼륨 그룹을 활성화하지 않아야 하며 다중 사용자 모드로 변경하지 않아야 합니다(예를 들어, /sbin/init 2를 지정하면 안 됨). 이를 따르지 않으면 루트 파일 시스템이 손상될 수 있습니다.

LVM 구성 정보를 복구 또는 복원한 경우 다음 명령을 사용하여 시스템을 다시 부팅합니다.

```
# /sbin/reboot
```

LVM 유지 관리 모드 부팅 및 LVM 구조 관련 문제 해결에 대한 자세한 내용은 **Disk and File Management Tasks on HP-UX**(1997년, Prentice Hall PTR 발행)를 참조하십시오.

로그 파일 및 추적 파일

LVM에는 전용 로그 파일이나 추적 파일이 없습니다. LVM은 오류나 경고를 /var/adm/syslog/syslog.log에 기록합니다.

I/O 오류

장치 드라이버가 I/O 요청에 대해 LVM에 오류를 반환하면 LVM은 이 오류를 **복구 가능** 또는 **복구 불가능**으로 분류합니다.

복구 가능한 오류

LVM에 복구 가능하거나 수정 가능한 오류가 발생한 경우에는 오류가 스스로 해결되거나 사용자가 오류를 해결하는 단계를 수행할 수 있다는 가정하에 실패한 작업을 내부적으로 다시 시도합니다. 복구 가능한 오류의 예는 다음과 같습니다.

- 장치 전원 장애
- **볼륨 그룹이 활성화된 후에** 누락된 디스크
- 누락된 디스크로 표시되는 느슨한 디스크 케이블

이 경우 LVM은 콘솔에 오류 메시지를 기록하지만 논리 볼륨에 액세스하는 응용 프로그램에 오류를 반환하지 않습니다.

별도의 작동 미러에 데이터의 현재 사본이 있는 경우에는 LVM은 복구 불가능한 오류에서와 마찬가지로 미러 사본에 I/O를 지시합니다. 논리 볼륨에 액세스하는 응용 프로그램은 오류를 감지하지 않습니다. 미러 간에 데이터 동기화를 유지하기 위해 LVM은 현재 사본이 다른 곳에 있더라도 문제가 있는 디스크에 복구 가능한 쓰기 요청을 다시 시도합니다. 그러나 이 프로세스는 LVM 내부의 데몬에 의해 관리되며 논리 볼륨에 대한 사용자 액세스에는 영향을 미치지 않습니다.

그러나 문제의 장치에 데이터 사본만 있는 경우 LVM은 성공할 때까지 즉, 장치가 응답하거나 시스템이 다시 부팅될 때까지 I/O 요청을 다시 시도합니다. 논리 볼륨에 I/O를 수행하는 응용프로그램이 차단되고 장치가 복구될 때까지 기다립니다. 이 경우 응용 프로그램 또는 파일 시스템이 정지하고 응답하지 않을 수 있습니다.

일시적으로 사용할 수 없는 장치

기본적으로 LVM은 복구 가능한 오류가 있는 I/O 요청을 성공할 때까지 또는 시스템이 다시 부팅될 때까지 재시도합니다. 그러므로 응용 프로그램이나 파일 시스템이 정지하면 콘솔 로그에서 디스크 드라이브의 문제점을 확인하고 실패한 장치를 서비스로 복원하는 조치를 취하는 방법으로 문제를 해결해야 합니다.

영구적으로 사용할 수 없는 장치

I/O 요청 재시도가 성공하지 않는 경우(예: 디스크가 실제로 제거된 경우)에는 응용 프로그램 또는 파일 시스템이 무제한 차단될 수 있습니다. 응용 프로그램이 응답하지 않으면 시스템을 다시 부팅해야 합니다.

다시 부팅의 대안으로서 논리 볼륨에서 제한 시간을 설정하여 복구 가능한 오류를 복구 불가능 오류로 처리하기 전에 LVM이 복구 가능한 오류를 재시도할 기간을 제어할 수 있습니다. 장치가 해당 시간 내에 응답하지 않으면 LVM은 호출자에게 I/O 오류를 반환합니다. 이 제한 시간 값은 기본 물리 볼륨 제한 시간 및 드라이버 제한 시간에 따라 달라지므로 LVM에서는 논리 볼륨 제한 시간이 만료되고 몇 초 후에 I/O 오류를 반환할 수 있습니다.

제한 시간 값은 일반적으로 0입니다. 이는 무제한 제한 시간을 의미합니다. 따라서 재시도에 성공할 때까지 호출자에게 I/O 요청을 반환하지 않습니다.

다음과 같이 `lvdisplay` 명령을 사용하여 논리 볼륨의 제한 시간 값을 확인합니다.

```
# lvdisplay /dev/vg00/lvol1 | grep Timeout
IO 시간 제한(초)          기본값
```

lvchange 명령의 -t 옵션을 사용하여 시간 제한 값을 설정합니다. 그러면 논리 볼륨에 대한 제한 시간 값(초 단위)이 설정됩니다. 예를 들어 /dev/vg01/lvol1의 제한 시간을 1분으로 설정하려면 다음 명령을 입력합니다.

```
# lvchange -t 60 /dev/vg01/lvol1
```



주의: 논리 볼륨에서 제한 시간을 설정하면 일시적인 오류를 복구 불가능 오류로 취급할 가능성이 높아지므로 논리 볼륨에서 읽고 쓰는 응용 프로그램에 I/O 오류가 발생할 수 있습니다. 응용 프로그램이 이러한 오류를 처리할 준비가 되지 않은 경우에는 무제한 논리 볼륨 제한 시간을 유지하십시오.



팁: 기본 물리 볼륨에 할당된 제한 시간의 정적분으로 논리 볼륨 제한 시간을 설정합니다. 그렇지 않으면 I/O 요청의 실제 지속 기간이 논리 볼륨 제한 시간을 초과할 수 있습니다. 물리 볼륨에서 I/O 제한 시간 값을 변경하는 방법에 대한 자세한 내용은 pvchange(1M)를 참조하십시오.

복구 불가능 오류

복구 불가능 오류는 치명적인 것으로 간주됩니다. 작업 재시도가 작동할 것으로 기대되지 않습니다.

별도의 작동 미러에 데이터의 현재 사본이 있는 경우에는 LVM에서 미러 사본에 읽기 및 쓰기를 지시합니다. 논리 볼륨에 액세스하는 응용 프로그램의 I/O 작업이 성공적으로 완료됩니다.

그러나 데이터의 다른 사본이 없는 경우에는 LVM에서 논리 볼륨에 액세스하는 모든 하위 시스템에 오류를 반환합니다. 따라서 논리 볼륨에 직접 액세스하는 응용 프로그램은 I/O 요청 실패에 대해 준비가 되어 있어야 합니다. VxFS와 같은 파일 시스템과 대부분의 데이터베이스 응용 프로그램은 오류 상황에서 복구할 수 있도록 설계되었습니다. 예를 들어 VxFS에 I/O 오류가 발생하면 파일 시스템에 대한 액세스 및 해당 파일의 하위 집합에 대한 액세스를 사용 불가능으로 설정합니다.

LVM에서는 다음과 같은 두 가지 상황을 복구 불가능으로 간주합니다.

미디어 오류

미디어 오류로 인해 I/O 요청이 실패한 경우 일반적으로 LVM은 오류 발생 시 콘솔 로그 파일(/var/adm/syslog/syslog.log)에 메시지를 인쇄합니다. 미디어 오류가 발생하면 디스크를 교체해야 합니다([“손상된 디스크 교체” \(126 페이지\)](#) 참조).

디스크 하드웨어가 손상된 블록 자동 배치 기능(일반적으로 하드웨어 스페어링이라고 함)을 지원할 경우에는 해당 기능을 활성화합니다. 이 기능은 LVM이 감지한 미디어 오류를 최소화합니다.



참고: LVM에서는 손상된 블록의 소프트웨어 배치를 수행하지 않습니다. LVM은 LVM의 이전 릴리즈에서 작성된 소프트웨어 배치 항목을 인식하여 사용하지만 새 항목을 만들지는 않습니다. `lvchange`를 사용하여 손상된 블록 배치를 설정하거나 해제하더라도 위와 같은 사항에는 아무 변동이 없습니다.

볼륨 그룹 활성화 시 장치 누락

볼륨 그룹을 활성화할 때 I/O 관련 장치가 없으면 LVM은 활성화할 때 사용자의 터미널에 오류 메시지를 인쇄합니다. 디스크를 찾아서 서비스 상태로 복원하거나 디스크를 교체한 다음 볼륨 그룹을 다시 활성화해야 합니다.

볼륨 그룹 활성화 장애

일반적으로 볼륨 그룹은 시스템이 시작하는 동안 자동으로 활성화됩니다. `vgchange`를 사용하여 의도적으로 볼륨 그룹을 비활성화하지 않는 한 볼륨 그룹을 수동으로 활성화할 필요가 없습니다. 모든 경우에 LVM에서는 볼륨 그룹의 디스크 쿼럼이 사용 가능한 상태여야 합니다.

쿼럼은 볼륨 그룹을 활성화하거나 볼륨 그룹을 활성 상태로 유지하기 위해 볼륨 그룹에서 사용 가능해야 하는 필수 물리 볼륨 수입니다. 볼륨 그룹을 활성화하려면 마지막 활성 상태에서 사용 가능했던 디스크 중 **절반 이상**이 온라인 및 서비스 상태여야 합니다. 볼륨 그룹이 완전히 작동 가능한 상태를 유지하려면 디스크의 **절반 이상**이 존재하고 사용 가능해야 합니다.

실행 시간 동안 볼륨 그룹이 활성화된 후 디스크에 장애가 발생하거나 오프라인 상태가 되면 쿼럼이 손실될 수 있습니다. 이러한 상황은 볼륨 그룹에 정의된 물리 볼륨의 반 미만만 현재 완전히 작동 상태로 남아 있는 경우 발생합니다. 예를 들어 볼륨 그룹에 디스크 두 개가 속해 있는 경우 하나가 손실되면 볼륨 그룹을 활성화하는 경우처럼 쿼럼이 손실되지는 않습니다. 두 디스크를 모두 사용할 수 없는 상태가 되어야 쿼럼이 손실됩니다. 이런 경우 볼륨 그룹은 여전히 활성화되어 있지만 볼륨 그룹에서 쿼럼이 손실되었다는 메시지가 콘솔에 출력됩니다. 쿼럼이 복원될 때까지(이전 예제에서 볼륨 그룹에 있는 LVM 디스크 중 적어도 하나가 다시 사용 가능하게 되면) LVM은 사용자가 볼륨 그룹 구성에 영향을 미치는 대부분의 명령을 완료할 수 없게 합니다. 또한 기본 디스크에 액세스할 수 없으므로 해당 볼륨 그룹의 논리 볼륨에 대한 일부 I/O 액세스가 중지될 수 있습니다. 마찬가지로 쿼럼이 복원될 때까지 LVM은 LVM 정보의 일관성(무결성)을 보장할 수 없으므로 MWC가 업데이트되지 않습니다.

`vgchange -q n` 옵션을 사용하여 볼륨 그룹이 활성화될 때 시스템의 쿼럼 검사를 재정의합니다. 이 옵션은 런타임 쿼럼 검사에 영향을 주지 않습니다. 쿼럼을 재정의하면 볼륨 그룹의 구성이 부정확해질 수 있습니다(예: 최근에 만든 논리 볼륨 누락). 이 구성 변경을 되돌릴 수 없을 수도 있습니다.

볼륨 그룹을 활성화할 때 또는 부팅할 때 쿼럼 요구 사항을 재정의하는 방법이 있습니다. LVM이 허용하는 경우라도 디스크 쿼럼이 없는 활성 볼륨 그룹에 대한 LVM 구성을 변경하지 않는 것이 좋습니다. 쿼럼 문제를 해결하려면 사용할 수 없는 디스크를 서비스 상태로 되돌리는 것이 좋습니다.

비 루트 볼륨 그룹의 쿼럼 문제

쿼럼을 설정하는 데 필요한 디스크가 부족할 때 비 루트 볼륨 그룹을 활성화하려고 하면 다음과 같은 오류 메시지가 vgchange에 표시됩니다.

```
# vgchange -a y /dev/vg01
vgchange: 경고: 볼륨 그룹 물리 볼륨
"/dev/dsk/c1t0d2"에 연결할 수 없습니다.
물리 볼륨 경로가 존재하지 않는 장치를 참조하거나
커널에 구성되어 있지 않습니다.
vgchange: 볼륨 그룹 "/dev/vg01"을(를) 활성화할 수 없습니다.
물리 볼륨이 연결되지 않았거나 유효한 VGDA가
물리 볼륨에 없습니다.
```

쿼럼에 문제가 발생하여 비 루트 볼륨 그룹이 활성화되지 않은 경우 다음 단계를 수행합니다.

1. 활성화할 수 없는 볼륨 그룹에 속한 모든 디스크의 데이터 연결 및 전원(Fibre Channel 구역화 및 보안 포함)을 확인합니다. 모든 디스크(또는 적어도 쿼럼을 만드는 데 필요한 디스크)를 서비스 상태로 되돌립니다. 그런 다음 vgchange 명령을 사용하여 볼륨 그룹을 다시 활성화합니다.
2. 쿼럼을 사용 가능한 상태로 만들 수 있는 방법이 없으면 vgchange 명령에 -q 옵션을 사용하여 쿼럼 요구 사항을 재정의합니다.

```
# vgchange -a y -q n /dev/vg01
```

볼륨 그룹이 쿼럼 없이 활성화됩니다. 논리 볼륨의 일부 또는 전체가 존재하지 않는 디스크 중 하나에 위치하므로 특정 논리 볼륨에 액세스할 수 없다는 메시지가 표시될 수 있습니다.

쿼럼 요구 사항을 재정의할 때마다 최신 상태가 아닌 데이터를 사용하게 될 위험이 있습니다. 데이터가 최신 상태인지 확인하려면 활성화된 볼륨 그룹에 있는 논리 볼륨의 데이터뿐만 아니라 논리 볼륨의 크기 및 위치도 확인합니다.

가능한 빨리 비활성화된 디스크를 볼륨 그룹으로 되돌립니다. 원래 볼륨 그룹을 활성화했을 때 온라인 상태가 아닌 디스크를 서비스 상태로 되돌리려면 다음과 같이 vgchange 명령을 사용합니다.

```
# vgchange -a y /dev/vg01
```

루트 볼륨 그룹의 쿼럼 문제

루트 볼륨 그룹에서도 쿼럼 문제가 보고될 수 있습니다. 루트 볼륨 그룹에 쿼럼을 구성할 만큼 디스크가 충분하지 않으면 부팅 과정에서 물리 볼륨이 부족하다는 메시지가 표시됩니다. 이 오류는 시스템에서 사용할 필요가 없어진 디스크를 물리적으로 제거했지만 vgreduce를 사용하여 볼륨 그룹에서 물리 볼륨을 제거하지 않은 경우 발생합니다. 볼륨 그룹에서 LVM 디스크를 먼저 제거하지 않고 시스템에서 제거하지 마십시오. 그러나 쿼럼 재정의 옵션 -lq를 사용하여 시스템을 부팅하면 복구할 수 있습니다.

HP 9000 서버에서 다음 명령을 입력합니다.

```
ISL> hpux -lq
```

HP Integrity 서버에서 다음 명령을 입력합니다.

```
HPUX> boot -lq
```

버전 2.x 볼륨 그룹 활성화 장애

쿼럼 부족으로 인해 버전 2.x 볼륨 그룹이 활성화되지 않을 수 있습니다. 예를 들어 vgchange에 다음과 같은 오류 메시지가 표시될 수 있습니다.

```
# vgchange -a y /dev/vgtest
vgchange: 경고: 볼륨 그룹 물리 볼륨
"/dev/disk/disk1"에 연결할 수 없습니다.
I/O 오류
vgchange: I/O 오류
vgchange:볼륨 그룹 "/dev/vgtest"을(를) 활성화할 수 없습니다.
쿼럼이 없거나 일부 물리 볼륨이 없습니다.
```

쿼럼에 문제가 발생하여 버전 2.x 볼륨 그룹이 활성화되지 않은 경우 “비 루트 볼륨 그룹의 쿼럼 문제” (121 페이지)에 설명된 단계를 수행합니다.

필요한 명령이나 커널 드라이버가 없는 경우에도 버전 2.x 볼륨 그룹이 활성화되지 않을 수 있습니다. 예를 들어 vgchange에 다음과 같은 오류 메시지가 표시될 수 있습니다.

```
# vgchange -a y /dev/vgtest
vgchange: 오류: "lvmp" 드라이버가 로드되지 않았습니다.
```

다음은 표시될 수 있는 다른 오류 메시지입니다.

```
# vgchange -a y /dev/vgtest
vgchange: 경고: 볼륨 그룹 물리 볼륨
"/dev/disk/disk1"에 연결할 수 없습니다.
잘못된 바이트 시퀀스입니다.
vgchange:볼륨 그룹 "/dev/vgtest"을(를) 활성화할 수 없습니다.
쿼럼이 없거나 일부 물리 볼륨이 없습니다.
```

다음은 표시될 수 있는 또 다른 오류 메시지입니다.

```
# vgchange -a y /dev/vgtest
vgchange: 경고: 볼륨 그룹 물리 볼륨
"/dev/dsk/c1t0d0"에 연결할 수 없습니다.
교차 장치 링크
vgchange: 경고: 물리 볼륨 "/dev/dsk/c1t0d0"을(를) 쿼리할 수 없습니다.
지정된 경로가 이 볼륨 그룹에 연결된 물리 볼륨에
해당하지 않습니다.
vgchange: 물리 볼륨 목록을 쿼리할 수 없습니다.
vgchange: 볼륨 그룹 "/dev/vgtest"을(를) 활성화할 수 없습니다.
쿼럼이 없거나 일부 물리 볼륨이 없습니다.
```

이러한 활성화 장애 문제를 해결하려면 다음과 같이 lvmdm 명령을 실행합니다.

```
# lvmdm -t -V 2.0
```

필요한 명령 및 드라이버가 있으면 lvmdm에 다음이 표시됩니다.

```
--- LVM 제한 ---
VG 버전                      2.0
최대 VG 크기(TB)             2048
최대 LV 크기(TB)             256
최대 PV 크기(TB)             16
최대 VG 수                   512
```

```

최대 LV 수                511
최대 PV 수                511
최대 미러 수              6
최대 스트라이프 수        511
최대 스트라이프 크기(KB)  262144
LV당 최대 LX 수           33554432
PV당 최대 PX 수           16777216
최대 확장 영역 크기(MB)   256

```

lvmadm에 출력이 표시되지 않으면 운영 체제 릴리즈에서 버전 2.x 볼륨을 지원하지 않는 것입니다. 시스템을 HP-UX 11i v3의 2008년 3월 릴리즈나 최신 릴리즈로 업데이트해야 합니다.

버전 2.x 볼륨 그룹을 지원하는 커널 드라이버가 로드되지 않으면 lvmadm에 다음 오류가 표시됩니다.

```
# lvmadm -t -V 2.0
lvmadm: 오류: "lvmp" 드라이버가 로드되지 않았습니다.
```

다음과 같이 kcmodule 명령을 사용하여 lvmp 모듈을 로드합니다.

```
# kcmodule lvmp=best
==> Update the automatic 'backup' configuration first? n
* Future operations will ask whether to update the backup.
* The requested changes have been applied to the currently
  running configuration.

Module          State   Cause      Notes
lvmp            (before)  unused     loadable, unloadable
                (now)      loaded     best
                (next boot) loaded     explicit
```

다시 부팅할 필요가 없습니다. lvmp 모듈이 로드되면 lvmadm이 성공합니다.

```
# lvmadm -t -V 2.0
--- LVM 제한 ---
VG 버전                2.0
최대 VG 크기(TB)       2048
최대 LV 크기(TB)       256
최대 PV 크기(TB)       16
최대 VG 수             512
최대 LV 수             511
최대 PV 수             511
최대 미러 수           6
최대 스트라이프 수      511
최대 스트라이프 크기(KB) 262144
LV당 최대 LX 수         33554432
PV당 최대 PX 수         16777216
최대 확장 영역 크기(MB) 256
```

루트 볼륨 그룹 검색

LVM 하위 시스템이 부팅 디스크에서 중요한 정보가 손상되었음을 감지하면 연결된 모든 장치를 검색하여 루트 볼륨 그룹에 속한 물리 볼륨을 찾으려고 시도합니다. 그러면 시스템 콘솔 및 /var/adm/syslog/syslog.log에 다음 메시지가 표시됩니다.

```
LVM : Failure in attaching PV (dev=0x10000nn) to the root volume group.
The physical volume does not belong to the root volume group
LVM : Failure in attaching PV (dev=0x10000nn) to the root volume group.
The physical volume does not belong to the root volume group
LVM : Activation of root volume group failed
Quorum not present, or some physical volume(s) are missing
LVM: Scanning for Root VG PVs (VGID 0xn timer 0xn timer)
```

이 루트 볼륨 그룹 검색에 성공하면 다음과 같은 메시지가 나타납니다.

```
LVM: Rootvgscan detected 10 PV(s). Will attempt root VG activation
using the following PV(s):
    0x100005f 0x1000060 0x1000061 0x1000062 0x1000063 0x1000064
    0x1000065 0x1000067 0x1000068 0x100006e
LVM: WARNING: Root VG activation required a scan. The PV information in
the on-disk BDRA may be out-of-date from the system's current IO
configuration. To update the on-disk BDRA, first update /etc/lvmtab
using vgscan(1M), then update the on-disk BDRA using lvmboot(1M).
For example, if the root VG name is /dev/vg00:
    1. vgscan -k -f /dev/vg00
    2. lvmboot -R /dev/vg00
LVM: Root VG activated
```

이 루트 볼륨 그룹 검색에서 모든 물리 볼륨을 찾지 못하면 다음 메시지가 나타납니다.

```
LVM: WARNING: Rootvgscan did not find any PV(s) matching root VGID.
Will attempt root VG activation using the boot device (0x10000nn).
```

또는 다음 메시지가 나타납니다.

```
LVM: WARNING: BDRA lists the number of PV(s) for the root VG as nn,
but rootvgscan found only nn. Proceeding with root VG activation.
```

LVM 부팅 장애

LVM 구성이 부팅되지 않는 데에는 몇 가지 이유가 있습니다. 비LVM 디스크의 부팅과 관련된 문제 외에도 LVM 기반 시스템이 부팅되지 않는 문제가 발생할 수 있습니다.

쿼럼 부족

이 시나리오에서 루트 볼륨 그룹에는 **쿼럼** 요구 사항에 맞는 디스크가 부족합니다. 부팅할 때 사용할 수 있는 물리 볼륨이 부족하다는 메시지가 표시됩니다.

```
panic: LVM: Configuration failure
```

루트 볼륨 그룹을 활성화하고 성공적으로 시스템을 부팅하려면 사용 가능한 LVM 디스크 수가 볼륨 그룹이 마지막으로 활성 상태였을 때 연결되었던 LVM 디스크 수의 절반 이상이어야 합니다. 따라서 마지막 활성화 과정에서 루트 볼륨 그룹에 두 개의 디스크가 연결된 경우, "절반 이상" 필요하다는 것은 두 디스크가 모두 사용 가능한 상태여야 함을 의미합니다. 쿼럼 장애를 처리하는 방법에 대한 자세한 내용은 “볼륨 그룹 활성화 장애”(120 페이지)를 참조하십시오.

디스크의 손상된 LVM 데이터 구조

LVM 부팅 가능 디스크는 BDRA에 중요한 부팅 정보를 포함하고 있습니다. 이 정보가 손상되었거나 최신 상태가 아니거나 더 이상 존재하지 않을 수 있습니다. BDRA에서 최

신 정보를 유지 관리하는 것은 중요하므로 루트, 부팅, 기본 스왑 또는 덤프 논리 볼륨의 위치에 영향을 주는 변경 사항이 발생할 때마다 `lvrmboot` 또는 `lvlnboot` 명령을 사용하십시오.

이 문제를 해결하려면 “유지 관리 모드 부팅” (116 페이지)의 설명대로 유지 관리 모드에서 시스템을 부팅한 다음 시스템 LVM 데이터 구조의 손상 내용을 복구해야 합니다. 부팅 디스크에서 `vgcfgrestore`를 사용하십시오.

손상된 LVM 구성 파일

볼륨 그룹 활성화와 관련된 다른 문제로 `/etc/lvmtab` 또는 `/etc/lvmtab_p` 파일이 누락되거나 손상될 수 있습니다. 유지 관리 모드로 부팅한 후 `vgscan` 명령을 사용하여 `/etc/lvmtab` 및 `/etc/lvmtab_p` 파일을 다시 만들 수 있습니다. 자세한 내용은 `vgscan(1M)`을 참조하십시오.

논리 볼륨의 크기를 축소시킨 후 발생하는 문제

파일 시스템이 논리 볼륨에 처음 만들어질 때 논리 볼륨이 허용하는 크기로 만들어집니다.

파일 시스템을 확장하지 않고 논리 볼륨을 확장한 경우 파일 시스템의 크기보다 큰 동안에는 논리 볼륨의 크기를 안전하게 축소시킬 수 있습니다. 파일 시스템의 크기를 확인하려면 `bdf` 명령을 사용합니다. 파일 시스템을 확장한 후에는 더 이상 연관된 논리 볼륨의 크기를 안전하게 축소시킬 수 없습니다.

`lvreduce` 명령을 사용하여 파일 시스템이 들어 있는 논리 볼륨의 크기를 파일 시스템의 크기보다 작게 축소시키면 파일 시스템이 손상됩니다. 그 다음에 손상된 파일 시스템을 마운트하려고 하면 시스템이 손상됩니다. 이런 경우 다음 단계를 수행하십시오.

1. 단일 사용자 모드로 시스템을 다시 부팅합니다.
2. 손상된 파일 시스템에 대한 최신 데이터 백업이 있으면 이 단계를 건너뛰십시오. 이러한 백업 데이터가 없고 해당 데이터가 중요한 경우 해당 파일 시스템의 파일을 백업하여 손상되지 않은 모든 데이터를 복구합니다.

최신 백업을 시도하기 전에 다음 사항을 고려합니다.

- 백업 프로그램이 파일 시스템의 손상된 부분에 액세스하면 시스템이 다시 손상됩니다. 다음 단계를 진행하려면 시스템을 다시 부팅해야 합니다.
- 해당 파일 시스템에 있는 데이터의 전체(또는 일부)가 손상되지 않았거나 복구 가능하다는 보장은 없습니다. 이 단계는 가능한 많이 복구하려는 시도입니다. 즉, 이 단계에서 성공적으로 백업된 데이터는 복구할 수 있지만 파일 손상으로 인해 데이터의 일부 또는 전체가 성공적으로 백업되지 않을 수 있습니다.

3. 손상된 파일 시스템이 마운트되어 있으면 즉시 마운트 해제합니다.
4. 스왑 공간 또는 원시 데이터 저장소에 논리 볼륨을 사용하거나 `HP SMH` 또는 `newfs` 명령을 사용하여 논리 볼륨에 새 파일 시스템을 만듭니다. 이 새로운 파일 시스템은 현재 크기가 축소된 논리 볼륨에 대응됩니다.

5. 논리 볼륨에 새로운 파일 시스템을 만든 경우 다음 중 하나를 수행합니다.
 - 이전에 백업한 내용이 있으면(2단계의 백업 아님) 내용을 복원합니다. 크기를 줄인 논리 볼륨의 새 파일 시스템은 원래 파일 시스템보다 크기가 작으므로 원래 파일을 모두 복원할 만큼 공간이 충분하지 않을 수 있습니다.
 - 이전에 백업한 내용이 없으면 2단계에서 만든 백업에서 최대한 많은 파일을 복원합니다.
 - 새로운 파일 시스템을 사용하여 새로운 파일 세트를 만들고 저장합니다(원래 파일을 복원하지 않음).

손상된 디스크 교체

디스크는 물리적 장치이므로 하드웨어에 장애가 발생하여 디스크를 교체해야 할 수 있습니다. 장애가 발생한 디스크를 새 디스크(혼동을 방지하기 위해 원래 디스크의 하드웨어 주소 유지)로 교체한 후 백업에서 해당 디스크로 데이터를 복원해야 합니다.

디스크가 LVM의 제어 아래 있으므로 디스크에 여러 개의 논리 볼륨에 대한 물리 확장 영역을 포함할 수 있습니다. 우선 해당 논리 볼륨의 레이아웃을 복원하고 해당 논리 볼륨 각각의 데이터를 백업으로부터 복원해야 합니다.

이 절에서는 장애가 발생한 LVM 디스크를 교체하는 단계별 지침을 제공합니다.

디스크에 장애가 발생하기 전에 수행해야 할 단계는 “[LVM 시스템 복구 준비](#)” (50 페이지)를 검토하십시오. 이 절을 신중하게 읽고 가능한 빨리 필요한 절차를 구현해야 합니다. 시스템 복구가 이러한 단계에 의존할 수 있습니다.

복구 과정에 대한 질문이 있으면 해당 지역 HP 고객 지원 센터에 문의하십시오.



팁: 디스크 장애에 대한 자세한 내용은 **When Good Disks Go Bad: Dealing with Disk Failures under LVM** 백서(<http://docs.hp.com>)를 참조하십시오. 이 백서에서는 디스크 장애 인식, 장애 디스크 식별 및 적절한 해결책 선택(예: 디스크 교체 대신 제거)과 같은 추가 항목에 대해 설명합니다. 또한 HP-UX 11i v3 이전 릴리즈도 다룹니다.

디스크 교체 전제 조건

장애가 발생한 디스크를 분리한 후 교체 과정은 다음 질문에 대한 대답에 따라 달라집니다.

• 핫 스왑 가능 디스크입니까?

전원이 켜져 있고 SCSI 버스가 활성 상태인 동안 시스템에서 비활성 상태의 핫 스왑 가능 하드 디스크 드라이브 모듈을 제거하거나 추가할 수 있습니다. 즉, 전체 시스템의 전원을 끄지 않고 시스템에서 핫 스왑 가능 디스크를 교체하거나 제거할 수 있습니다.

시스템에서 핫 스왑 가능 디스크에 대한 정보는 시스템 하드웨어 설명서를 참조하십시오. 기타 하드 디스크의 사양은 <http://docs.hp.com>의 해당 설치 안내서를 참조하십시오.

핫 스왑 가능 디스크가 아닌 경우 디스크를 교체하려면 시스템 종료 시간을 예약해야 합니다.

- **디스크가 루트 디스크 또는 루트 볼륨 그룹의 일부입니까?**

루트 디스크에 장애가 발생하면 교체 과정에 부팅 영역을 설정하는 단계가 포함됩니다. 또한 기본 루트 디스크에 장애가 발생하면 미러를 통해 부팅해야 합니다. 장애가 발생한 루트 디스크가 미러링되어 있지 않으면 교체 디스크로 다시 설치하거나 Ignite-UX 백업에서 복구해야 합니다.

디스크가 루트 볼륨 그룹에 있는지 확인하려면 `lvlnboot` 명령에 `-v` 옵션을 사용합니다. 루트 볼륨 그룹의 디스크 및 디스크에 구성된 특수 볼륨을 나열합니다. 예를 들면 다음과 같습니다.

```
# lvlnboot -v
볼륨 그룹 /dev/vg00에 대한 부팅 정의:
루트 볼륨 그룹에 속하는 물리 볼륨:
    /dev/disk/disk47_p2 -- 부팅 디스크
부팅: lv011      커밋: /dev/disk/disk47_p2
루트: lv013      커밋: /dev/disk/disk47_p2
스왑: lv012      커밋: /dev/disk/disk47_p2
덤프: lv012      커밋: /dev/disk/disk47_p2, 0
```

- **디스크에 있는 논리 볼륨이란 무엇이고 해당 볼륨은 미러링되어 있습니까?**

디스크를 교체한 후 백업에서 데이터를 복원해야 할 수 있습니다. 그러나 볼륨 그룹에 있는 논리 볼륨 하위 집합의 데이터만 복구하면 됩니다. 실제로 디스크에 물리 확장 영역이 있는 논리 볼륨에만 영향을 미칩니다. 또한 논리 볼륨이 미러링되어 있으면 미래에 최신 데이터 사본이 있을 수 있으므로 백업에서 복구할 필요가 없습니다.

`pvdisk` 명령으로 디스크를 사용하는 논리 볼륨 목록을 찾을 수 있습니다. `-v` 옵션을 사용하면 `pvdisk`는 물리 볼륨에 있는 모든 물리 확장 영역 목록 및 해당 확장 영역이 속한 논리 볼륨을 표시합니다. 이 목록은 너무 길기 때문에 목록을 `more`로 파이프하거나 파일로 보내야 합니다. 예를 들면 다음과 같습니다.

```
# pvdisk -v /dev/disk/disk3 | more
...
--- 물리 볼륨 배포 ---
LV 이름              LV의 LE    LV에 대한 PE
/dev/vg00/lvol5      50         50
/dev/vg00/lvol6      245        245
...
```

이 예제에서 논리 볼륨 `/dev/vg00/lvol5` 및 `/dev/vg00/lvol6`의 이 디스크에는 물리 확장 영역이 있으므로 `lvol5` 및 `lvol6`만 복원해야 합니다.

`pvdisk`가 실패하면 미리 작성한 구성 문서를 참조하거나 HP 지원 담당자가 제공하는 `vgcfgdisplay` 명령을 사용할 수 있습니다.

영향을 받는 각 논리 볼륨의 경우 `lvdisplay`를 사용하여 미러 사본 수가 0보다 큰지 확인합니다. 이 명령은 논리 볼륨이 미러링되어 있는지 확인합니다. 예를 들면 다음과 같습니다.

```
# lvsdisplay /dev/vg00/lvol1
--- 논리 볼륨 ---
LV 이름                /dev/vg00/lvol1
VG 이름                /dev/vg00
LV 권한              읽기/쓰기
LV 상태              사용 가능/동기화됨
미러 복사본          1
일관성 복구          MWC
일정                  병렬
LV 크기 (MB)          300
현재 LE              75
할당된 PE             150
스트라이프            0
스트라이프 크기 (KB)  0
손상된 블록          깨짐
할당                  엄격/연속
IO 시간 제한 (초)     기본값
```

미러 사본 수가 0이 아니므로 논리 볼륨이 미러링되어 있습니다.

lvsdisplay를 다시 사용하여 의심스러운 디스크로 매핑된 논리 확장 영역 및 **다른 디스크에 있는** 해당 데이터의 최신 사본이 있는지 여부를 확인합니다. -v 옵션을 사용하면 lvsdisplay에는 모든 논리 확장 영역, 물리 확장 영역으로의 매핑 및 해당 물리 확장 영역의 상태(못 쓰게 됨 또는 최신)가 표시됩니다.

이 목록은 너무 길 수 있으므로 grep를 사용하여 목록을 교체할 디스크로 제한합니다. 예를 들면 다음과 같습니다.

```
# lvsdisplay -v /dev/vg00/lvol1 | grep -e /dev/disk/disk3 -e '???'
00000 /dev/disk/disk3 00000 현재 /dev/disk/disk6 00000 현재
00001 /dev/disk/disk3 00001 현재 /dev/disk/disk6 00001 현재
00002 /dev/disk/disk3 00002 현재 /dev/disk/disk6 00002 현재
00003 /dev/disk/disk3 00003 현재 /dev/disk/disk6 00003 현재
00004 /dev/disk/disk3 00004 현재 /dev/disk/disk6 00004 현재
00005 /dev/disk/disk3 00005 현재 /dev/disk/disk6 00005 현재
...
```

이 예제에서 /dev/disk/disk3에 있는 lvol1 물리 확장 영역은 모두 시스템의 다른 위치, 특히 /dev/disk/disk6에 최신 사본이 있습니다. 볼륨 그룹이 활성화 될 때 /dev/disk/disk3을 사용할 수 없으면 해당 열에 디스크 이름 대신 '???'가 포함됩니다.

수집된 정보에 따라 적절한 절차를 선택합니다.

미러된 부팅할 수 없는 디스크 교체

디스크의 물리 확장 영역이 **모두** 다른 디스크에 사본이 있고 디스크가 부팅 디스크가 **아닌** 경우 이 절차를 사용합니다. 디스크에 미러되지 않은 논리 볼륨 또는 사용 가능한 최신 미러 사본이 없는 미러 논리 볼륨이 들어 있으면 “미러되지 않은 부팅할 수 없는 디스크 교체” (131 페이지)를 참조하십시오.

이 예제에서 교체할 디스크는 /dev/disk/disk14 및 /dev/rdisk/disk14라는 장치 특수 파일과 함께 LUN 하드웨어 경로 0/1/1/1.0x3.0x0에 있습니다. 다음 단계를 수행하십시오.

1. 하드웨어 경로를 디스크에 저장합니다.

ioscan 명령을 실행하고 장애가 발생한 디스크의 하드웨어 경로를 기록합니다.

```
# ioscan -m lun /dev/disk/disk14
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0    esdisk  CLAIMED   DEVICE   offline HP MSA Vol
        0/1/1/1.0x3.0x0
                /dev/disk/disk14      /dev/rdisk/disk14
```

이 예제에서 LUN 인스턴스 번호는 14, LUN 하드웨어 경로는 64000/0xfa00/0x0이며 LUN 경로 하드웨어 경로는 0/1/1/1.0x3.0x0입니다.

장애가 발생한 디스크가 교체되면 새로운 LUN 인스턴스와 LUN 하드웨어 경로가 만들어집니다. 교체된 디스크를 식별하려면 LUN 하드웨어 경로(0/1/1/1.0x3.0x0)를 사용해야 합니다.

2. 디스크에 대한 LVM 액세스를 중지합니다.

핫 스왑 가능 디스크가 아닌 경우 디스크를 교체하려면 시스템 전원을 끕니다. 시스템을 종료하면 디스크에 대한 LVM 액세스가 중지되므로 이 단계를 건너뛸 수 있습니다.

핫 스왑 가능 디스크인 경우 pvchange 명령의 -a 옵션을 사용하여 분리합니다.

```
# pvchange -a N /dev/disk/disk14
```

3. 디스크를 교체합니다.

디스크를 교체하는 방법에 대한 하드웨어 세부 사항은 시스템 또는 디스크 어레이에 대한 하드웨어 관리 설명서를 참조하십시오.

핫 스왑 가능 디스크인 경우 해당 디스크를 교체합니다.

핫 스왑 가능 디스크가 아닌 경우에는 시스템을 종료하고 전원을 끈 다음 디스크를 교체합니다. 시스템을 다시 부팅합니다.

4. 디스크가 교체되었음을 대용량 저장소 하위 시스템에 알립니다.

장애가 발생한 디스크를 교체하기 위해 시스템을 다시 부팅하지 않은 경우 새 디스크를 기존 디스크의 교체용으로 사용하기 전에 scsimgr를 실행합니다. 예를 들면 다음과 같습니다.

```
# scsimgr replace_wwid -D /dev/rdisk/disk14
```

이 명령을 사용하면 저장소 하위 시스템이 기존 디스크의 LUN WWID(World-Wide-Identifier)를 새 디스크의 LUN WWID로 바꿀 수 있습니다. 저장소 하위 시스템은 교체 디스크에 대해 새로운 LUN 인스턴스와 장치 특수 파일을 만듭니다.

5. 교체 디스크의 새 LUN 인스턴스 번호를 확인합니다.

예를 들면 다음과 같습니다.

```
# iotscan -m lun
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0  esdisk  NO_HW     DEVICE   offline HP MSA Vol
              /dev/disk/disk14      /dev/rdisk/disk14
...
disk 28 64000/0xfa00/0x1c esdisk  CLAIMED   DEVICE   online  HP MSA Vol
              0/1/1/1.0x3.0x0
              /dev/disk/disk28      /dev/rdisk/disk28
```

이 예제에서는 LUN 하드웨어 경로 64000/0xfa00/0x1c, 장치 특수 파일 /dev/disk/disk28 및 /dev/rdisk/disk28과 함께 새 디스크에 대한 LUN 인스턴스 28을 만들어 기존 디스크와 동일한 LUN 경로 하드웨어 경로 0/1/1/1.0x3.0x0에 있습니다. 기존 디스크에 대한 기존 LUN 인스턴스 14에는 더 이상 연관된 LUN 경로가 없습니다.



참고: 장애가 발생한 디스크를 교체하기 위해 시스템을 다시 부팅한 경우에는 `iotscan -m lun`에 기존 디스크가 표시되지 않습니다.

6. 기존 인스턴스 번호를 교체 디스크에 할당합니다.

예를 들면 다음과 같습니다.

```
# io_redirect_dsf -d /dev/disk/disk14 -n /dev/disk/disk28
```

그러면 기존 LUN 인스턴스 번호(14)가 교체 디스크에 할당됩니다. 또한 새 디스크의 장치 특수 파일 이름이 기존 LUN 인스턴스 번호와 일관성을 갖도록 바꿉니다. 다음 `iotscan -m lun` 출력에서 결과를 보여 줍니다.

```
# iotscan -m lun /dev/disk/disk14
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x1c esdisk  CLAIMED   DEVICE   online  HP MSA Vol
              0/1/1/1.0x3.0x0
              /dev/disk/disk14      /dev/rdisk/disk14
```

LUN 하드웨어 경로 64000/0xfa00/0x0이 있는 기존 디스크의 LUN 표현이 제거되었습니다. LUN 하드웨어 경로 64000/0xfa00/0x1c가 있는 새 디스크의 LUN 표현이 LUN 인스턴스 28에서 LUN 인스턴스 14로 다시 할당되었으며 해당 장치 특수 파일의 이름은 /dev/disk/disk14 및 /dev/rdisk/disk14로 바뀌었습니다.

7. LVM 구성 정보를 새 디스크에 복원합니다.

예를 들면 다음과 같습니다.

```
# vgcfgrestore -n /dev/vgmn /dev/rdisk/disk14
```

8. 디스크에 대한 LVM 액세스를 복원합니다.

단계 2에서 시스템을 다시 부팅하지 않은 경우 다음과 같이 디스크를 다시 연결합니다.

```
# pvchange -a y /dev/disk/disk14
```

시스템을 다시 부팅한 경우 다음과 같이 볼륨 그룹을 다시 활성화하여 디스크를 다시 연결합니다.

```
# vgchange -a y /dev/vgmn
```



참고: 비활성화되었거나 이미 활성화된 볼륨 그룹에서 `vgchange` 명령에 `-a y` 옵션을 실행할 수 있습니다. 볼륨 그룹의 모든 디스크에 대한 모든 경로를 연결하고 볼륨 그룹에서 오프라인인 디스크나 교체된 디스크의 자동 복구를 다시 시작합니다. 따라서 볼륨 그룹의 모든 디스크 및 경로에서 모든 작업이 완료된 후에만 `vgchange`를 실행하여 모두 연결해야 합니다.

교체된 디스크의 모든 데이터가 미러링되므로 다른 작업을 수행할 필요가 없습니다. LVM은 디스크의 데이터를 다른 데이터 미러 사본과 자동으로 동기화합니다.

미러되지 않은 부팅할 수 없는 디스크 교체

디스크의 물리 확장 영역이 **모두** 다른 위치에 미러 사본이 없고 사용 중인 디스크가 부팅 디스크가 아닌 경우 이 절차를 사용합니다.

이 예제에서 교체할 디스크는 `/dev/disk/disk14` 및 `/dev/rdisk/disk14`라는 장치 특수 파일과 함께 LUN 하드웨어 경로 `0/1/1/1.0x3.0x0`에 있습니다. 다음 단계를 수행하십시오.

1. 하드웨어 경로를 디스크에 저장합니다.

`ioscan` 명령을 실행하고 장애가 발생한 디스크의 하드웨어 경로를 기록합니다.

```
# ioscan -m lun /dev/disk/disk14
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0    esdisk  CLAIMED   DEVICE   offline HP MSA Vol
        0/1/1/1.0x3.0x0
                /dev/disk/disk14      /dev/rdisk/disk14
```

이 예제에서 LUN 인스턴스 번호는 14, LUN 하드웨어 경로는 `64000/0xfa00/0x0`이며 LUN 경로 하드웨어 경로는 `0/1/1/1.0x3.0x0`입니다.

장애가 발생한 디스크가 교체되면 새로운 LUN 인스턴스와 LUN 하드웨어 경로가 만들어집니다. 교체된 디스크를 식별하려면 LUN 하드웨어 경로(`0/1/1/1.0x3.0x0`)를 사용해야 합니다.

2. 디스크에 대한 LVM 액세스를 중지합니다.

핫 스왑 가능 디스크가 아닌 경우 디스크를 교체하려면 시스템 전원을 끕니다. 시스템을 종료하면 디스크에 대한 LVM 액세스가 중지되므로 이 단계를 건너뛸 수 있습니다.

핫 스왑 가능 디스크인 경우 미러되지 않은 모든 논리 볼륨에 대한 사용자 및 LVM 액세스를 비활성화합니다.

우선 모든 미러되지 않은 논리 볼륨에 대한 **사용자** 액세스를 비활성화합니다. 모든 응용 프로그램을 중지하고 이러한 논리 볼륨을 사용하는 파일 시스템을 마운트 해제합니다. 이렇게 하면 응용 프로그램 또는 파일 시스템이 새로 복원된 교체 디스크에 일치하지 않는 데이터를 쓸 수 없습니다.

해당 디스크를 사용하는 미러되지 않은 각 논리 볼륨에 대해 다음을 수행하십시오.

- a. `fuser` 명령을 사용하여 원시 장치 또는 파일 시스템으로 논리 볼륨에 액세스하는 사용자가 있는지 확인합니다. 파일 시스템에 열려 있는 파일이 있거나 사용자의 현재 작업 디렉토리인 경우 `fuser`는 사용자의 프로세스 ID를 보고합니다.

예를 들어 논리 볼륨이 `/dev/vg01/lvol1`이면 다음 명령을 입력합니다.

```
# fuser -cu dev/vg01/lvol1
/dev/vg01/lvol1:    27815c(root)    27184c(root)
```

- b. `fuser`가 논리 볼륨을 사용하는 프로세스 ID를 보고할 경우 `ps` 명령을 사용하여 프로세스 ID 목록을 프로세스에 매핑한 다음 해당 프로세스를 중지할 수 있는지 여부를 확인합니다.

예를 들어 다음과 같이 프로세스 27815 및 27184를 찾습니다.

```
# ps -fp27815 -p27184
      UID      PID  PPID  C      STIME TTY          TIME COMMAND
    root  27815  27184   0  09:04:05 pts/0        0:00 vi test.c
    root  27184  27182   0  08:26:24 pts/0        0:00 -sh
```

- c. 중지할 수 있으면 `fuser`에 `-k` 옵션을 사용하여 논리 볼륨에 액세스하는 모든 프로세스를 강제 종료합니다.

이 예제 프로세스는 중요하지 않으므로 다음과 같이 강제 종료합니다.

```
# fuser -ku dev/vg01/lvol1
/dev/vg01/lvol1:    27815c(root)    27184c(root)
```

- d. 논리 볼륨을 파일 시스템으로 사용하려면 다음과 같이 마운트 해제합니다.

```
# umount /dev/vg01/lvol1
```



참고: 논리 볼륨을 사용하여 응용 프로그램을 중지할 수 없거나 파일 시스템을 마운트 해제할 수 없는 경우에는 시스템을 종료해야 합니다.

미러되지 않은 논리 볼륨에 대한 사용자 액세스를 비활성화한 후 디스크에 대한 LVM 액세스를 비활성화합니다.

```
# pvchange -a N /dev/disk/disk14
```

3. 디스크를 교체합니다.

디스크를 교체하는 방법에 대한 하드웨어 세부 사항은 시스템 또는 디스크 어레이에 대한 하드웨어 관리 설명서를 참조하십시오.

핫 스왑 가능 디스크인 경우 해당 디스크를 교체합니다.

핫 스왑 가능 디스크가 아닌 경우에는 시스템을 종료하고 전원을 끈 다음 디스크를 교체합니다. 시스템을 다시 부팅합니다.

4. 디스크가 교체되었음을 대용량 저장소 하위 시스템에 알립니다.

장애가 발생한 디스크를 교체하기 위해 시스템을 다시 부팅하지 않은 경우 새 디스크를 기존 디스크의 교체용으로 사용하기 전에 `scsimgr`를 실행합니다. 예를 들면 다음과 같습니다.

```
# scsimgr replace_wwid -D /dev/rdisk/disk14
```

이 명령을 사용하면 저장소 하위 시스템이 기존 디스크의 LUN WWID(World-Wide-Identifier)를 새 디스크의 LUN WWID로 바꿀 수 있습니다. 저장소 하위 시스템은 교체 디스크에 대해 새로운 LUN 인스턴스와 장치 특수 파일을 만듭니다.

5. 교체 디스크의 새 LUN 인스턴스 번호를 확인합니다.

예를 들면 다음과 같습니다.

```
# ioscan -m lun
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0  esdisk  NO_HW      DEVICE   offline HP MSA Vol
              /dev/disk/disk14      /dev/rdisk/disk14
...
disk 28 64000/0xfa00/0x1c  esdisk  CLAIMED    DEVICE   online  HP MSA Vol
              0/1/1/1.0x3.0x0
              /dev/disk/disk28      /dev/rdisk/disk28
```

이 예제에서는 LUN 하드웨어 경로 64000/0xfa00/0x1c, 장치 특수 파일 /dev/disk/disk28 및 /dev/rdisk/disk28과 함께 새 디스크에 대한 LUN 인스턴스 28을 만들어 기존 디스크와 동일한 LUN 경로 하드웨어 경로 0/1/1/1.0x3.0x0에 있습니다. 기존 디스크에 대한 기존 LUN 인스턴스 14에는 더 이상 연관된 LUN 경로가 없습니다.



참고: 장애가 발생한 디스크를 교체하기 위해 시스템을 다시 부팅한 경우에는 `ioscan -m lun`에 기존 디스크가 표시되지 않습니다.

6. 기존 인스턴스 번호를 교체 디스크에 할당합니다.

예를 들면 다음과 같습니다.

```
# io_redirect_dsf -d /dev/disk/disk14 -n /dev/disk/disk28
```

그러면 기존 LUN 인스턴스 번호(14)가 교체 디스크에 할당됩니다. 또한 새 디스크의 장치 특수 파일 이름이 기존 LUN 인스턴스 번호와 일관성을 갖도록 바꿉니다. 다음 `ioscan -m lun` 출력에서 결과를 보여 줍니다.

```
# ioscan -m lun /dev/disk/disk14
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x1c  esdisk  CLAIMED    DEVICE   online  HP MSA Vol
              0/1/1/1.0x3.0x0
              /dev/disk/disk14      /dev/rdisk/disk14
```

LUN 하드웨어 경로 64000/0xfa00/0x0이 있는 기존 디스크의 LUN 표현이 제거되었습니다. LUN 하드웨어 경로 64000/0xfa00/0x1c가 있는 새 디스크의 LUN 표현이 LUN 인스턴스 28에서 LUN 인스턴스 14로 다시 할당되었으며 해당 장치 특수 파일의 이름은 /dev/disk/disk14 및 /dev/rdisk/disk14로 바뀌었습니다.

7. LVM 구성 정보를 새 디스크에 복원합니다.

예를 들면 다음과 같습니다.

```
# vgcfgrestore -n /dev/vgmn /dev/rdisk/disk14
```

8. 디스크에 대한 LVM 액세스를 복원합니다.

단계 2에서 시스템을 다시 부팅하지 않은 경우 다음과 같이 디스크를 다시 연결합니다.

```
# pvchange -a y /dev/disk/disk14
```

시스템을 다시 부팅한 경우 다음과 같이 볼륨 그룹을 다시 활성화하여 디스크를 다시 연결합니다.

```
# vgchange -a y /dev/vgmn
```



참고: 비활성화되었거나 이미 활성화된 볼륨 그룹에서 `vgchange` 명령에 `-a y` 옵션을 실행할 수 있습니다. 볼륨 그룹의 모든 디스크에 대한 모든 경로를 연결하고 볼륨 그룹에서 오프라인인 디스크나 교체된 디스크의 자동 복구를 다시 시작합니다. 따라서 볼륨 그룹의 모든 디스크 및 경로에서 모든 작업이 완료된 후에만 `vgchange`를 실행하여 모두 연결해야 합니다.

9. 손실된 데이터를 복구합니다.

LVM은 디스크에서 미리 논리 볼륨을 모두 복구하고 볼륨 그룹이 활성화될 때 해당 복구를 시작합니다.

단계 2에서 식별된 미러되지 않은 모든 논리 볼륨의 경우 다음과 같이 백업에서 데이터를 복원하고 사용자 액세스를 다시 활성화합니다.

- 원시 볼륨의 경우 백업을 만드는 데 사용된 유틸리티를 사용하여 전체 원시 볼륨을 복원합니다. 그런 다음 응용 프로그램을 다시 시작합니다.
- 파일 시스템의 경우 우선 파일 시스템을 다시 만들어야 합니다. 예를 들면 다음과 같습니다.

```
# newfs -F fstype /dev/vgmn/rlvolnn
```

`newfs` 명령에 논리 볼륨의 특성 장치 파일을 사용합니다. 기본이 아닌 구성이 있는 파일 시스템의 경우 올바른 옵션에 대해서는 `newfs(1M)`를 참조하십시오.

파일 시스템을 만든 후 이전에 차지한 마운트 지점에 마운트합니다. 그런 다음 전체 백업에서 해당 파일 시스템의 데이터를 복원합니다.



팁: 파일 시스템 재작성 단계를 더 간단히 하려면 원래 파일 시스템을 만든 방법을 기록합니다. 파일 시스템 성능 조정에 사용된 매개 변수와 같은 기타 파일 시스템 매개 변수를 변경할 수 있습니다. 파일 시스템은 디스크 장애가 발생하기 전보다 용량이 커야 합니다.

미러 부팅 디스크 교체

미러 부팅 디스크를 교체할 경우 다음과 같은 두 가지 추가 작업을 수행해야 합니다.

1. 교체 디스크에 대한 부팅 정보를 초기화해야 합니다.
2. 교체할 때 시스템을 다시 부팅해야 하고 기본 부팅 디스크가 교체 중이면 대체 부팅 디스크에서 부팅해야 합니다.

이 예제에서 교체할 디스크는 `/dev/disk/disk14` 및 `/dev/rdisk/disk14`라는 장치 특수 파일과 함께 LUN 하드웨어 경로 `0/1/1/1.0x3.0x0`에 있습니다. 시스템이 HP Integrity 서버이므로 물리 볼륨 이름은 부팅 디스크의 HP-UX 파티션을 지정해야 합니다(`/dev/disk/disk14_p2` 및 `/dev/disk/disk14_p2`).

1. 하드웨어 경로를 디스크에 저장합니다.

`ioscan` 명령을 실행하고 다음과 같이 장애가 발생한 디스크의 하드웨어 경로를 기록합니다.

```
# ioscan -m lun /dev/disk/disk14
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0    esdisk CLAIMED  DEVICE   offline HP MSA Vol
        0/1/1/1.0x3.0x0
                /dev/disk/disk14      /dev/rdisk/disk14
                /dev/disk/disk14_p1 /dev/rdisk/disk14_p1
                /dev/disk/disk14_p2 /dev/rdisk/disk14_p2
                /dev/disk/disk14_p3 /dev/rdisk/disk14_p3
```

이 예제에서 LUN 인스턴스 번호는 14, LUN 하드웨어 경로는 `64000/0xfa00/0x0`이며 LUN 경로 하드웨어 경로는 `0/1/1/1.0x3.0x0`입니다.

장애가 발생한 디스크가 교체되면 새로운 LUN 인스턴스와 LUN 하드웨어 경로가 만들어집니다. 교체된 디스크를 식별하려면 LUN 하드웨어 경로(`0/1/1/1.0x3.0x0`)를 사용해야 합니다.

2. 디스크에 대한 LVM 액세스를 중지합니다.

핫 스왑 가능 디스크가 아닌 경우 디스크를 교체하려면 시스템 전원을 끕니다. 시스템을 종료하면 디스크에 대한 LVM 액세스가 중지되므로 이 단계를 건너뛸 수 있습니다.

핫 스왑 가능 디스크인 경우 `pvchange` 명령의 `-a` 옵션을 사용하여 장치를 분리합니다.

```
# pvchange -a N /dev/disk/disk14_p2
```



참고: HP 9000 서버에서는 부팅 디스크가 파티션되지 않으므로 물리 볼륨은 HP-UX 파티션이 아닌 전체 디스크를 가리킵니다. 다음 명령을 사용합니다.

```
# pvchange -a N /dev/disk/disk14
```

3. 디스크를 교체합니다.

디스크를 교체하는 방법에 대한 하드웨어 세부 사항은 시스템 또는 디스크 어레이에 대한 하드웨어 관리 설명서를 참조하십시오.

핫 스왑 가능 디스크인 경우 해당 디스크를 교체합니다.

핫 스왑 가능 디스크가 아닌 경우에는 시스템을 종료하고 전원을 끈 다음 디스크를 교체합니다. 시스템을 다시 부팅합니다. 다음과 같은 두 가지 문제가 발생할 수 있습니다.

- 일반적으로 부팅하는 데 사용하는 디스크를 교체하면 부팅 로더가 필요로 하는 정보가 교체 디스크에 포함되지 않습니다. 이 경우 부팅 과정이 중단되고 대체 부팅 경로로 구성되어야 하는 미러 부팅 디스크에서 부팅됩니다.
- 루트 볼륨 그룹에 디스크가 두 개만 있으면 “볼륨 그룹 활성화 장애” (120 페이지)에서 설명한 대로 시스템이 쿼럼 검사에 실패할 수 있습니다. 부팅 과정 초기에 다음 메시지와 함께 장애가 발생할 수 있습니다.

```
panic: LVM: Configuration failure
```

이 경우 성공적으로 부팅하려면 쿼럼을 재정의해야 합니다. 부팅 과정을 중단하고 부팅 명령에 `-lq` 옵션을 추가하여 이 작업을 수행합니다.

부팅 과정 및 부팅 옵션 선택 방법에 대한 자세한 내용은 **HP-UX 시스템 관리 설명서: 구성 관리**를 참조하십시오.

4. 디스크가 교체되었음을 대용량 저장소 하위 시스템에 알립니다.

장애가 발생한 디스크를 교체하기 위해 시스템을 다시 부팅하지 않은 경우 새 디스크를 기존 디스크의 교체용으로 사용하기 전에 `scsimgr`를 실행합니다. 예를 들면 다음과 같습니다.

```
# scsimgr replace_wwid -D /dev/rdisk/disk14
```

이 명령을 사용하면 저장소 하위 시스템이 기존 디스크의 LUN WWID(World-Wide-Identifier)를 새 디스크의 LUN WWID로 바꿀 수 있습니다. 저장소 하위 시스템은 교체 디스크에 대해 새로운 LUN 인스턴스와 장치 특수 파일을 만듭니다.

5. 교체 디스크의 새 LUN 인스턴스 번호를 확인합니다.

예를 들면 다음과 같습니다.

```
# ioscan -m lun
Class I Lun H/W Path          Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x0 esdisk NO_HW      DEVICE  offline HP MSA Vol
          /dev/disk/disk14          /dev/rdisk/disk14
          /dev/disk/disk14_p1  /dev/rdisk/disk14_p1
          /dev/disk/disk14_p2  /dev/rdisk/disk14_p2
```



```

/dev/disk/disk14_p3 /dev/rdisk/disk14_p3
...
disk 28 64000/0xfa00/0x1c esdisk CLAIMED DEVICE online HP MSA Vol
0/1/1/1.0x3.0x0
/dev/disk/disk28 /dev/rdisk/disk28

```

이 예제에서는 LUN 하드웨어 경로 64000/0xfa00/0x1c, 장치 특수 파일 /dev/disk/disk28 및 /dev/rdisk/disk28과 함께 새 디스크에 대한 LUN 인스턴스 28을 만들어 기존 디스크와 동일한 LUN 경로 하드웨어 경로 0/1/1/1.0x3.0x0에 있습니다. 기존 디스크에 대한 기존 LUN 인스턴스 14에는 더 이상 연관된 LUN 경로가 없습니다.



참고: 장애가 발생한 디스크를 교체하기 위해 시스템을 다시 부팅한 경우에는 `ioscan -m lun`에 기존 디스크가 표시되지 않습니다.

6. (HP Integrity 서버만 해당)교체 디스크를 파티션합니다.

“HP Integrity 서버에서 부팅 디스크 미리방” (101 페이지)에 설명된 대로 `idisk` 명령 및 파티션 설명 파일을 사용하여 디스크를 파티션하고 `insf`를 사용하여 파티션 장치 파일을 만듭니다.

7. 기존 인스턴스 번호를 교체 디스크에 할당합니다.

예를 들면 다음과 같습니다.

```
# io_redirect_dsf -d /dev/disk/disk14 -n /dev/disk/disk28
```

그러면 기존 LUN 인스턴스 번호(14)가 교체 디스크에 할당됩니다. 또한 새 디스크의 장치 특수 파일 이름이 기존 LUN 인스턴스 번호와 일관성을 갖도록 바꿉니다. 다음 `ioscan -m lun` 출력에서 결과를 보여 줍니다.

```
# ioscan -m lun /dev/disk/disk14
Class I Lun H/W Path      Driver  S/W State H/W Type Health  Description
=====
disk 14 64000/0xfa00/0x1c esdisk  CLAIMED  DEVICE   online  HP MSA Vol
0/1/1/1.0x3.0x0
      /dev/disk/disk14      /dev/rdisk/disk14
      /dev/disk/disk14_p1  /dev/rdisk/disk14_p1
      /dev/disk/disk14_p2  /dev/rdisk/disk14_p2
      /dev/disk/disk14_p3  /dev/rdisk/disk14_p3

```

LUN 하드웨어 경로 64000/0xfa00/0x0이 있는 기존 디스크의 LUN 표현이 제거되었습니다. LUN 하드웨어 경로 64000/0xfa00/0x1c가 있는 새 디스크의 LUN 표현이 LUN 인스턴스 28에서 LUN 인스턴스 14로 다시 할당되었으며 해당 장치 특수 파일의 이름은 /dev/disk/disk14 및 /dev/rdisk/disk14로 바뀌었습니다.

8. LVM 구성 정보를 새 디스크에 복원합니다.

예를 들면 다음과 같습니다.

```
# vgcfgrestore -n /dev/vg00 /dev/rdisk/disk14_p2
```



참고: HP 9000 서버에서는 부팅 디스크가 파티션되지 않으므로 물리 볼륨은 HP-UX 파티션이 아닌 전체 디스크를 가리킵니다. 다음 명령을 사용합니다.

```
# vgcfgrestore -n /dev/vg00 /dev/rdisk/disk14
```

9. 디스크에 대한 LVM 액세스를 복원합니다.

단계 2에서 시스템을 다시 부팅하지 **않은 경우** 다음과 같이 디스크를 다시 연결합니다.

```
# pvchange -a y /dev/disk/disk14_p2
```

HP 9000 서버에서 이 명령을 사용합니다.

```
# pvchange -a y /dev/disk/disk14
```

시스템을 다시 부팅한 경우 다음과 같이 볼륨 그룹을 다시 활성화하여 디스크를 다시 연결합니다.

```
# vgchange -a y /dev/vg00
```



참고: 비활성화되었거나 이미 활성화된 볼륨 그룹에서 vgchange 명령에 -a y 옵션을 실행할 수 있습니다. 볼륨 그룹의 모든 디스크에 대한 모든 경로를 연결하고 볼륨 그룹에서 오프라인인 디스크나 교체된 디스크의 자동 복구를 다시 시작합니다. 따라서 볼륨 그룹의 모든 디스크 및 경로에서 모든 작업이 완료된 후에만 vgchange를 실행하여 모두 연결해야 합니다.

10. 디스크에서 부팅 정보를 초기화합니다.

HP Integrity 서버의 경우 “HP Integrity 서버에서 부팅 디스크 미러링” (101 페이지)의 5단계와 6단계 설명에 따라 부팅 영역을 설정하고 디스크의 EFI 파티션에서 자동 부팅 파일을 업데이트합니다.

HP 9000 서버의 경우 “HP 9000 서버에서 부팅 디스크 미러링” (99 페이지)의 4단계와 5단계 설명에 따라 부팅 영역을 설정하고 자동 부팅 파일을 업데이트합니다.

미러되지 않은 부팅 디스크 교체

미러되지 않은 부팅 디스크에 장애가 발생하면 시스템 부팅에 필요한 유일한 정보 사본이 손실됩니다. 교체 디스크로 다시 설치하거나 Ignite-UX 백업에서 복구해야 합니다.

경고 및 오류 메시지

이 절에서는 LVM이 보고한 경고 및 오류 메시지의 일부를 나열합니다. 각 메시지에 대한 원인이 설명되고 작업이 권장됩니다.

물리 디스크 및 볼륨 그룹과 오류 메시지 일치

일반적으로 오류 메시지에 장치 파일 이름 대신 장치 번호가 들어 있습니다. 예를 들어 /var/adm/syslog/syslog.log에 다음 메시지가 표시될 수 있습니다.

```
SCSI: 요청 제한 시간 -- lbolt: 329741615, dev: 1f022000
```

오류 메시지를 특정 디스크에 매핑하려면 /dev 디렉토리에서 출력 값과 일치하는 장치 번호를 갖는 장치 파일을 찾습니다. 즉, 보조 번호가 dev: 뒤에 나오는 숫자의 뒤쪽 6자리와 일치하는 파일을 검색합니다. 이 예제에서 장치 번호는 1f022000입니다. 해당 번호의 뒤쪽 6자리는 022000이므로 다음 명령을 사용하여 해당 값을 검색합니다.

```
# ll /dev/*dsk | grep 022000
brw-r----- 1 bin      sys      31 0x022000 Sep 22  2002 c2t2d0
crw-r----- 1 bin      sys     188 0x022000 Sep 25  2002 c2t2d0
```

다음과 같이 pvdisplay 명령을 사용하여 이 물리 볼륨이 들어 있는 볼륨 그룹을 확인합니다.

```
# pvdisplay /dev/dsk/c2t2d0 | grep "VG Name"
VG 이름                /dev/vg03
```

pvdisplay 명령이 실패하면 다음과 같이 LVM 구성 파일 /etc/lvmtab 및 /etc/lvmtab_p에서 물리 볼륨을 검색합니다.

```
# lvmdm -l -V 1.0
--- 버전 1.0 볼륨 그룹 ---
VG 이름 /dev/vg00
PV 이름 /dev/disk/disk36_p2
```

```
VG 이름 /dev/vgtest
PV 이름 /dev/disk/disk43
PV 이름 /dev/disk/disk44
```

```
VG 이름 /dev/vg03
PV 이름 /dev/dsk/c2t2d0
```

lvmdm 버전에서 -l 옵션을 인식하지 못하면 다음과 같이 strings 명령을 사용합니다.

```
# strings /etc/lvmtab | more
/dev/vg00
/dev/disk/disk36_p2
/dev/vgtest
/dev/disk/disk43
/dev/disk/disk44
/dev/vg03
/dev/dsk/c2t2d0
```

이러한 명령의 출력에 따라 오류 메시지에서 볼륨 그룹 vg03에 속하는 물리 볼륨 /dev/dsk/c2t2d0을 나타냅니다.

마찬가지로 /var/adm/syslog/syslog.log의 일부 LVM 오류 메시지에는 볼륨 그룹의 장치 번호가 포함됩니다. 예를 들면 다음과 같습니다.

```
LVM: VG 128 0x002000: Lost quorum.
```

주 번호 128은 버전 2.x 볼륨 그룹임을 나타냅니다. 버전 1.0 볼륨 그룹의 주 번호는 64입니다. 이 오류 메시지를 볼륨 그룹에 매핑하려면 /dev 디렉토리에서 주 번호와 보조 번호가 일치하는 장치 번호를 갖는 볼륨 그룹 장치 파일을 검색합니다. 이 예제에서 주 번호는 128이고 보조 번호는 0x002000이므로 다음 명령을 사용하여 해당 값을 검색합니다.

```
# ll /dev/*/group | grep 128.0x002000
crw-r----- 1 root sys 128 0x002000 Jan  7 08:27 /dev/vgtest2/group
```

예제 오류 메시지는 버전 2.x 볼륨 그룹 vgtest를 나타냅니다.

모든 LVM 명령에 대한 메시지

메시지 텍스트

vgcfgbackup: /etc/lvmtab이(가) 실행 중인 커널과 맞지 않습니다.
커널은 "/dev/vgname"에 대해 #개 디스크를 나타내는데 /etc/lvmtab은(는) #개 디스크를 가
집니다.
백업을 계속할 수 없습니다.

원인

현재 물리 볼륨 수(Cur PV)와 활성 물리 볼륨(Act PV)이 다릅니다. 볼륨 그룹의 현재
PV와 활성 PV는 항상 일치해야 합니다. 또한 이 오류는 물리 볼륨을 볼륨 그룹에 일치시
키는 데 사용되는 /etc/lvmtab 또는 /etc/lvmtab_p의 LVM 데이터 구조가 메모리
와 디스크에서 오래되었음을 나타냅니다.

권장 작업

누락된 디스크를 찾습니다. 볼륨 그룹의 각 디스크에 대해 ioscan 및 diskinfo를 사
용하여 디스크가 제대로 작동 중인지 확인합니다.

lvchange(1M)

메시지 텍스트

"m": 잘못된 옵션입니다.

원인

시스템에 HP MirrorDisk/UX가 설치되지 않았습니다.

권장 작업

HP MirrorDisk/UX를 설치합니다.

lvextend(1M)

메시지 텍스트

lvextend: 사용 가능한 물리 확장 영역이 충분하지 않습니다.
논리 볼륨 "/dev/vgname/lvname"을(를) 확장할 수 없습니다.
엄격한 할당 정책에 의해 오류가 발생한 것 같습니다.

원인

볼륨 그룹에서 논리 볼륨을 요청된 크기로 확장할 공간이 부족합니다. 일반적으로 다음
과 같은 경우 중 하나로 인해 발생합니다.

1. 볼륨 그룹에 사용 가능한 물리 확장 영역이 부족합니다. `vgdisplay`를 실행하여 사용 가능한 물리 확장 영역 수를 확인하고 해당 숫자에 확장 영역 크기를 곱하여 볼륨 그룹에서 사용 가능한 공간을 확인합니다. 예를 들면 다음과 같습니다.

```
# vgdisplay vg00
--- 볼륨 그룹 ---
VG 이름                /dev/vg00
VG 쓰기 권한          읽기/쓰기
VG 상태               사용 가능
최대 LV                255
현재 LV                10
열린 LV                10
최대 PV                16
열린 PV                1
활성 PV                1
PV당 최대 PE           4350
VGDA                   2
PE 크기 (MB)           4
총 PE                  4340
할당된 PE              3740
사용 가능한 PE         600
총 PVG                 0
총 스페어 PV           0
사용 중인 총 스페어 PV  0
VG 버전                1.0
VG 최대 크기           1082g
VG 최대 확장 영역       69248
```

이 예제에서 총 사용 가능한 공간은 4MB 또는 2400MB의 물리 확장 영역 600개입니다.

2. 논리 볼륨이 엄격한 할당 정책에 따라 미러링되는데 할당 정책에 맞는 확장 영역이 개별 디스크에서 부족합니다. 이 내용을 확인하려면 `lvdisplay`를 실행하여 논리 볼륨이 차지하는 디스크를 확인한 다음 볼륨 그룹의 다른 디스크에 충분한 공간이 있는지 확인합니다.
3. SAN 환경에서 디스크 중 하나의 크기가 동적으로 증가했습니다. LVM이 크기의 비동기 변경을 감지하지 못했습니다.

권장 작업

1. 더 작은 크기의 논리 볼륨을 선택하거나 볼륨 그룹에 디스크 공간을 추가합니다.
2. 더 작은 크기의 논리 볼륨을 선택하거나 볼륨 그룹에 디스크 공간을 추가합니다. 또는 `pvmove`를 사용하여 사용 가능한 디스크의 공간을 해제합니다.
3. `vgmodify` 명령을 사용하여 디스크 크기 변경을 감지하고 새 공간을 볼륨 그룹에 통합합니다.

메시지 텍스트

"m": 잘못된 옵션입니다.

원인

시스템에 HP MirrorDisk/UX가 설치되지 않았습니다.

권장 작업

HP MirrorDisk/UX를 설치합니다.

lvlnboot(1M)

메시지 텍스트

lvlnboot: 스왑 논리 볼륨을 구성할 수 없습니다.
스왑 논리 볼륨 크기가 IODC 최대 주소를 초과합니다.

원인

부팅 디스크 펌웨어가 스왑 논리 볼륨의 전체 범위에 액세스할 수 없습니다. 이 문제는 기본 스왑이 디스크에서 4GB 넘게 구성된 경우 호스트 버스 어댑터에서 발생합니다.

권장 작업

시스템 펌웨어를 업그레이드하거나 블록 주소 지정을 지원하는 최신 호스트 버스 어댑터를 사용합니다. 이러한 작업에 모두 실패하면 기본 스왑 논리 볼륨의 크기를 4GB를 초과하지 않도록 축소시킵니다.

pvchange(1M)

메시지 텍스트

제공된 경로 이름을 통해 경로 또는 물리 볼륨을 분리할 수 없습니다.
pvchange(1M) -a N을 사용하여 연결된 경로를 사용하는 PV를 분리하거나
pvchange(1M) -a n을 사용하여 각 PV 경로를 개별적으로 분리하십시오.

원인

지정된 경로가 자신이 속한 활성 볼륨 그룹에 성공적으로 연결되지 않았으므로 해당 경로는 볼륨 그룹의 일부가 아닙니다.

권장 작업

지정된 경로 이름이 올바른지 확인합니다. 물리 볼륨을 분리하는 동안 오류가 발생한 경우 다른 경로를 지정합니다. 이전에 연결된 경로가 확실하지 않으면 pvchange에 -a n 옵션을 사용하여 물리 볼륨에 대한 각 경로를 각각 분리합니다.

메시지 텍스트

경고: 물리 볼륨을 분리하면 해당 디스크에 있는
논리 볼륨 내 데이터 가용성이 줄어듭니다.
물리 볼륨 또는 마지막으로 사용 가능한 물리 볼륨의 경로를 분리하기 전에
볼륨 그룹의 다른 디스크에서 사용 가능한 데이터의

대체 복사본이 있는지 확인하십시오. 필요한 경우 `pvchange (1M)`를 사용하여 이 작업을 취소하십시오.

원인

이 경고는 권장 사항이며 경로 또는 물리 볼륨이 분리될 때마다 생성됩니다.

권장 작업

없습니다.

vgcfgbackup(1M)

메시지 텍스트

물리 볼륨 `pvname`의 LVMREC가 잘못되었습니다.

원인

디스크의 LVM 헤더가 잘못되었습니다. 이 문제는 `dd` 또는 `pvcreate`와 같은 명령으로 기존 LVM 디스크를 덮어쓸 경우 발생할 수 있습니다. 두 시스템 간에 디스크를 공유하면 해당 디스크가 볼륨 그룹에 이미 있다는 것을 한 시스템에서 인식하지 못할 수 있습니다. BC 분리 볼륨을 사용할 때 `vgchgid`를 잘못 실행하면 손상이 발생할 수 있습니다.

권장 작업

`vgcfgrestore`를 사용하여 상태가 좋은 알려진 구성을 디스크에 복원합니다. 문제가 처음 발생하기 전에 날짜가 지정된 유효한 사본을 사용해야 합니다. 예를 들면 다음과 같습니다.

```
# vgcfgrestore -n vname pvname
```

vgcfgrestore(1M)

메시지 텍스트

물리 볼륨 `pvname`을(를) 복원할 수 없습니다.
PV를 복원하기 전에 먼저 PV를 분리하거나 VG를 비활성화하십시오.

원인

`vgcfgrestore` 명령이 이미 활성 볼륨 그룹에 속한 디스크를 초기화하는 데 사용되었습니다.

권장 작업

물리 볼륨을 복원하기 전에 물리 볼륨을 분리하거나 볼륨 그룹을 비활성화합니다. 디스크가 손상된 경우 디스크를 분리하고 `vgcfgrestore`를 사용하여 표시한 다음 디스크를 교체하지 않고 다시 연결합니다. 이로 인해 LVM이 디스크를 다시 초기화하고 해당 위치에 매핑된 미리 사용자 데이터를 동기화합니다.

vgchange(lM)

메시지 텍스트

vgchange: 경고: "lvmp" 드라이버가 로드되지 않았습니다.

원인

버전 2.x 볼륨 그룹을 활성화하고 있으며 버전 2.x 볼륨 그룹의 커널 드라이버는 로드되지 않습니다.

권장 작업

다음과 같이 lvmp 드라이버를 로드합니다.

```
# kcmodule lvmp=best
==> Update the automatic 'backup' configuration first? n
* Future operations will ask whether to update the backup.
* The requested changes have been applied to the currently
  running configuration.
```

Module		State	Cause	Notes
lvmp	(before)	unused		loadable, unloadable
	(now)	loaded	best	
	(next boot)	loaded	explicit	

메시지 텍스트

vgchange: 경고: 볼륨 그룹 물리 볼륨
"pvname"에 연결할 수 없습니다.
잘못된 바이트 시퀀스입니다.
vgchange: 볼륨 그룹 "vgname"을(를) 활성화할 수 없습니다.
쿼럼이 없거나 일부 물리 볼륨이 없습니다.

원인

버전 2.x 볼륨 그룹을 활성화하고 있으며 운영 체제 릴리즈에서 버전 2.x 볼륨을 지원하지 않습니다.

권장 작업

시스템을 HP-UX 11i v3의 2008년 3월 릴리즈나 최신 릴리즈로 업데이트합니다.

메시지 텍스트

경고: 물리 볼륨 "pvname"을(를) 쿼리할 수 없습니다.
지정된 경로가 이 볼륨 그룹에 연결된 물리 볼륨에
해당하지 않습니다.
물리 볼륨 목록을 쿼리할 수 없습니다.

원인

이 오류의 가능한 이유는 다음과 같습니다.

1. 볼륨 그룹이 활성화될 때 디스크가 누락되지만 나중에 복원되었습니다. 이 문제는 일반적으로 시스템이 다시 부팅되거나, 누락되거나 케이블이 연결되지 않았거나 전원이 꺼진 디스크를 통해 볼륨 그룹을 활성화한 경우 발생합니다.
2. 잘못된 볼륨 그룹 정보로 디스크 LVM 헤더를 덮어썼습니다. 두 시스템 간에 디스크를 공유하면 해당 디스크가 볼륨 그룹에 이미 있다는 것을 한 시스템에서 인식하지 못할 수 있습니다. 확인하려면 HP 지원 담당자가 제공하는 `dump_lvmtab` 명령을 사용하여 볼륨 그룹 정보를 확인하고 불일치를 찾습니다. 예를 들면 다음과 같습니다.

```
# dump_lvmtab -s | more
SYSTEM : 0x35c8cf58
TIME   : 0x3f9acc69 : Sat Oct 25 15:18:01 2003
FILE    : /etc/lvmtab
HEADER  : version:0x03e8   vgnum:7
VG[00]  VGID:35c8cf58 3dd13164 (@0x00040c) pvnum:2 state:0 /dev/vg00
        (00) VGID:35c8cf58 3dd13164 PVID:35c8cf58 3dd13164 /dev/dsk/c0t6d0
        (01) VGID:35c8cf58 3dd13164 PVID:35c8cf58 3dda4694 /dev/dsk/c4t6d0
VG[01]  VGID:065f303f 3e63f01a (@0x001032) pvnum:92 state:0 /dev/vg01
        (00) !VGID:35c8cf58 3f8df316 PVID:065f303f 3e63effa /dev/dsk/c40t0d0
        (01) !VGID:35c8cf58 3f8df316 PVID:065f303f 3e63effe /dev/dsk/c40t0d4
        (02) !VGID:35c8cf58 3f8df316 PVID:065f303f 3e63f003 /dev/dsk/c40t1d0
...
```

이 예제에서 `/dev/vg01`에 있는 디스크의 VGID가 일치하지 않습니다. 불일치는 `!VGID`로 표시됩니다.

권장 작업

1. `ioscan` 및 `diskinfo`를 사용하여 디스크가 제대로 작동 중인지 확인합니다. 다음 명령을 사용하여 볼륨 그룹을 다시 활성화합니다.

```
# vgchange -a y vgname
```

2. 이 오류에서 복구하는 데는 여러 가지 방법이 있습니다. 다음 절차에 요약된 명령이 이해되지 않으면 HP 지원 담당자에게 문의하십시오.

- a. `vgcfgrestore`를 사용하여 상태가 좋은 알려진 구성을 디스크에 복원합니다. 문제가 처음 발생하기 전에 날짜가 지정된 유효한 사본을 사용해야 합니다. 예를 들면 다음과 같습니다.

```
# vgcfgrestore -n vgname pvname
```

- b. 볼륨 그룹 및 해당 논리 볼륨을 다시 만들어서 가장 최신 백업에서 데이터를 복원합니다. “볼륨 그룹 만들기” (60 페이지) 및 “논리 볼륨 만들기” (63 페이지)를 참조하십시오.

- c. “볼륨 그룹 내보내기” (69 페이지) 및 “볼륨 그룹 가져오기” (70 페이지)의 설명에 따라 볼륨 그룹을 내보내고 다시 가져옵니다. 예를 들면 다음과 같습니다.

```
# vgexport -m vgname.map -v -f vgname.file /dev/vgname
# vgimport -m vgname.map -v -f vgname.file /dev/vgname
```

메시지 텍스트

vgchange: 볼륨 그룹 `"/dev/vgname"`의 고유 ID를 설정할 수 없습니다.

원인

동일한 보조 번호를 갖는 LVM 그룹 파일이 여러 개 있습니다.

권장 작업

LVM 그룹 파일을 나열합니다. 중복 보조 번호가 있으면 영향을 받는 볼륨 그룹 중 하나를 내보내고 선택적으로 고유 보조 번호를 갖는 새 그룹 파일을 만든 후 해당 볼륨 그룹을 다시 가져옵니다. 이 절차가 이해되지 않으면 HP 지원 담당자에게 문의하십시오.

```
# ll /dev/*/group
# vgexport -m vgname.map -v -f vgname.file /dev/vgname
# mkdir /dev/vgname
# mknod /dev/vgname/group c 64 unique_minor_number
# vgimport -m vgname.map -v -f vgname.file /dev/vgname
```

vgcreate(1M)

메시지 텍스트

vgcreate: `"/dev/vgname/group"`: 문자 장치가 아닙니다.

원인

볼륨 그룹 장치 파일이 없습니다. 이 버전의 `vgcreate` 명령에서는 해당 파일을 자동으로 만들지 않습니다.

권장 작업

[“볼륨 그룹 장치 파일 만들기” \(60 페이지\)](#)의 설명에 따라 볼륨 그룹에 대한 디렉토리를 만들고 그룹 파일을 만듭니다.

메시지 텍스트

vgcreate: 볼륨 그룹 `"/dev/vgname"`을(를) 만들 수 없습니다.
오류: 물리 볼륨 `"pvname"`에 BDRA가 포함되어 있습니다.
이 버전은 부팅 가능한 디스크를 지원하지 않으므로 볼륨 그룹 `"/dev/vgname"`에 추가할 수 없습니다.

원인

물리 볼륨 `pvname`은 부팅 가능한 디스크이며 `vgname`은 버전 2.x 볼륨 그룹입니다. 버전 2.x 볼륨 그룹은 부팅 가능한 물리 볼륨을 지원하지 않습니다.

권장 작업

[“LVM 사용을 위한 디스크 초기화” \(59 페이지\)](#)의 설명에 따라 `-B` 옵션 없이 `pvcreate` 명령을 사용하여 디스크를 다시 초기화합니다. 예를 들면 다음과 같습니다.

```
# pvcreate -f pvname
```

메시지 텍스트

`vgcreate`: 볼륨 그룹 `"/dev/vgname"`을(를) 만들 수 없습니다.
디스크에 대한 `VGRA`가 지정된 매개 변수에 대해 너무 큼니다.
확장 영역 크기를 늘리거나 `max_PV/max_LV`를 줄이고 다시 시도하십시오.

원인

각 LVM 디스크의 앞 부분에 있는 `VGRA`(Volume Group Reserved Area)가 볼륨 그룹의 디스크에 대한 일부 정보를 포함할 수 없습니다. 이 오류는 일반적으로 100GB가 넘는 디스크를 사용할 경우 발생합니다.

권장 작업

볼륨 그룹 작성 매개 변수를 조정합니다. `vgcreate` 명령의 `-s` 옵션을 사용하여 4MB가 넘는 확장 영역 크기를 선택하거나 `-p` 옵션을 사용하여 더 적은 물리 볼륨 수를 선택합니다. 이러한 옵션에 대한 자세한 내용은 `vgcreate(1M)`를 참조하십시오. 확장 영역 크기에 대한 권장 사항은 부록 C (161 페이지)를 참조하십시오.

`vgdisplay(1M)`

메시지 텍스트

`vgdisplay`: 볼륨 그룹 `"/dev/vgname"`을(를) 쿼리할 수 없습니다.
볼륨 그룹 보조 번호에 오류가 있을 수 있습니다.
그룹 보조 번호가 고유한지 확인하십시오.
`vgdisplay`: 볼륨 그룹 `"/dev/vgname"`을(를) 표시할 수 없습니다.

원인

이 오류의 가능한 이유는 다음과 같습니다.

1. 동일한 보조 번호를 갖는 LVM 그룹 파일이 여러 개 있습니다.
2. `Serviceguard`가 이전에 시스템에 설치되었고 `/dev/slvmvg` 장치 파일이 계속 존재합니다.

권장 작업

1. LVM 그룹 파일을 나열합니다. 중복 보조 번호가 있으면 영향을 받는 볼륨 그룹 중 하나를 내보내고 선택적으로 고유 보조 번호를 갖는 새 그룹 파일을 만든 후 해당 볼륨 그룹을 다시 가져옵니다. 이 절차가 이해되지 않으면 HP 지원 담당자에게 문의하십시오.

```
# ll /dev/*/group
# vgexport -m vgname.map -v -f vgname.file /dev/vgname
# mkdir /dev/vgname
# mknod /dev/vgname/group c 64 unique_minor_number
# vgimport -m vgname.map -v -f vgname.file /dev/vgname
```

2. `/dev/slvmvg` 장치 파일을 제거하고 다음 명령을 사용하여 `/etc/lvmtab` 및 `/etc/lvmtab_p` 파일을 다시 만듭니다.

```
# rm /dev/slvmvg
# mv /etc/lvmtab /etc/lvmtab.old
# mv /etc/lvmtab_p /etc/lvmtab_p.old
# vgscan -v
```

메시지 텍스트

```
vgdisplay: /etc/lvmtab: 해당 파일 또는 디렉토리가 없습니다.
vgdisplay: "/etc/lvmtab"에서 볼륨 그룹 이름을 읽을 수 없습니다.
vgdisplay: /etc/lvmtab: 해당 파일 또는 디렉토리가 없습니다.
vgdisplay: "/etc/lvmtab"에서 볼륨 그룹 이름을 읽을 수 없습니다.
```

원인

LVM 구성 파일 /etc/lvmtab 또는 /etc/lvmtab_p 중 하나가 누락되었습니다.

권장 작업

다음 명령을 사용하여 /etc/lvmtab 및 /etc/lvmtab_p 파일을 만듭니다.

```
# vgscan -v
```

vgscan 명령 및 옵션에 대한 자세한 내용은 [vgscan\(1M\)](#)을 참조하십시오.

메시지 텍스트

```
경고: 물리 볼륨 "pvname"을(를) 쿼리할 수 없습니다.
지정된 경로가 이 볼륨 그룹에 연결된 물리 볼륨에
해당하지 않습니다.
물리 볼륨 목록을 쿼리할 수 없습니다.
```

원인

이 오류의 가능한 이유는 “[vgchange\(1M\)](#)” (144 페이지) 오류 메시지에 설명되어 있습니다.

권장 작업

“[vgchange\(1M\)](#)” (144 페이지) 오류 메시지에서 권장 작업을 참조하십시오.

vgextend(1M)

메시지 텍스트

```
vgextend: 물리 볼륨당 물리 확장 영역이 충분하지 않습니다.
필요한 수: #, 소유 수: #.
```

원인

디스크 크기가 볼륨 그룹의 최대 디스크 크기를 초과합니다. 볼륨 그룹이 작성될 때 [vgcreate](#)의 -s 옵션으로 지정된 확장 영역 크기와 -e 옵션으로 지정된 디스크당 최대 물리 확장 영역 수의 곱으로 이 제한이 정의됩니다. 일반적으로 디스크는 볼륨 그룹에 성공적으로 추가되지만 일부 디스크는 액세스할 수 없습니다.

권장 작업

`vgmodify` 명령을 사용하여 디스크당 최대 물리 확장 영역 수를 조정합니다. 또는 `-s` 및 `-e` 옵션에 새 값을 사용하여 볼륨 그룹을 다시 만들 수 있습니다.

vgimport(1M)

메시지 텍스트

`vgimport: "/dev/vgname/group": 문자 장치가 아닙니다.`

원인

볼륨 그룹 장치 파일이 없습니다. 이 버전의 `vgimport` 명령에서는 해당 파일을 자동으로 만들지 않습니다.

권장 작업

“볼륨 그룹 장치 파일 만들기” (60 페이지)의 설명에 따라 볼륨 그룹에 대한 디렉토리를 만들고 그룹 파일을 만듭니다.

메시지 텍스트

볼륨 그룹 `/dev/vgname`에 있는 각 디스크의 고유 LVM 디스크 ID를 확인하지 못했습니다.

원인

이 메시지에에는 다음과 같은 두 가지 가능한 이유가 있습니다.

1. `vgimport` 명령에 `-s` 옵션을 사용했고 시스템에 있는 두 개 이상의 디스크에 동일한 LVM 식별자가 있습니다. 이 문제는 디스크를 BC 사본으로 만들거나 `dd`로 복제할 경우 발생할 수 있습니다.
2. LVM이 디스크 헤더를 읽지 못했습니다. 이 문제는 SAN 어레이에 대해 새 논리 단위를 만들 때 발생할 수 있습니다.

권장 작업

1. `-s` 옵션 없이 `vgimport`를 사용합니다. 또는 복사되거나 복제된 디스크에서 `vgchgid`를 사용하여 LVM 식별자를 변경합니다.
2. `vgimport` 명령을 다시 시도합니다.

/var/adm/syslog/syslog.log

메시지 텍스트

LVM: VG mm 0xnn0000:

Data in one or more logical volumes on PV nn 0x0nn000
was lost when the disk was replaced.

This occurred because the disk contained the only copy of the data.
Prior to using these logical volumes, restore the data from backup.

원인

LVM이 방금 교체된 디스크에 상주하는 미래되지 않은 논리 볼륨을 발견한 경우처럼 LVM이 교체된 디스크의 데이터를 자동으로 동기화할 수 없습니다.

권장 작업

논리 볼륨 내용을 백업에서 복원합니다.

메시지 텍스트

```
LVM: VG mm 0xnn0000: PVLink nn 0x0nn000 Detached.
```

이유:

이 메시지는 권장 사항이며 디스크 경로가 분리될 때마다 생성됩니다.

권장 작업

없습니다.

메시지 텍스트

```
LVM: vg[nn] pv[nn] No valid MCR, resyncing all mirrored MWC LVs on the PV
```

원인

이 메시지는 이전 릴리즈의 HP-UX에서 볼륨 그룹을 가져올 때 표시됩니다. HP-UX 11i v3에서 MWC의 형식이 변경되었으므로 볼륨 그룹에 MWC를 사용하는 미래 논리 볼륨이 들어 있으면 가져올 때 LVM에서 MWC를 변환합니다. 또한 모든 미래 논리 볼륨에 대해 완전 재동기화를 수행합니다. 이 작업에는 시간이 오래 걸릴 수 있습니다.

권장 작업

없습니다.

메시지 텍스트

```
LVM: vg 64 0xn timer: 장치  
0xn timer에서 이벤트 알림에 등록할 수 없음 (1)
```

원인

이 메시지는 HP-UX 11i v3으로 업그레이드한 후 처음으로 시스템을 부팅할 때 표시될 수 있습니다. 이 메시지는 I/O 구성 업데이트로 인해 표시되는 임시 메시지입니다. 나중에 부팅 프로세스에서 LVM이 이벤트 알림에 다시 등록되고 나면 문제가 해결됩니다.

권장 작업

없습니다.

문제 보고

LVM을 사용하여 문제를 해결할 수 없으면 다음 단계를 수행합니다.

1. **HP-UX Logical Volume Manager and MirrorDisk/UX Release Notes**를 읽어보고 알려진 문제인지 확인합니다. 알려진 문제일 경우 제공된 해결 방법에 따라 문제를 해결합니다.
2. 제품이 아직 보증 기간 중인지, 구입한 회사에서 해당 제품에 대한 서비스를 지원하는지 확인합니다. 운영 관리자로부터 필요한 정보를 제공받을 수 있습니다.
3. <http://www.itrc.hp.com>에 액세스하고 Technical Knowledge Database를 검색하여 현재 발생한 문제가 이미 보고되었는지 확인합니다. 액세스할 수 있는 설명서 및 리소스 종류는 자격 수준에 따라 달라집니다.



참고: ITRC 리소스 포럼(<http://www.itrc.hp.com>)에서는 피어 투 피어 지원을 통해 문제를 해결하며 등록된 사용자는 무료로 이용할 수 있습니다.

새로운 문제이거나 추가적인 도움말이 필요한 경우에는 <http://www.itrc.hp.com>의 지원 사례 관리자를 통해 온라인으로 또는 HP 지원 센터에 전화하여 HP Response Center에 문제를 기록합니다. 보증이 만료되었거나 올바른 지원 계약이 없는 경우에는 문제를 해결하는 데 필요한 시간 및 자재에 따라 유료로 지원 서비스를 받을 수 있습니다.

4. 문제와 관련된 정보를 제공하라는 메시지가 표시되면 요청된 정보를 수집하여 제출합니다.

A LVM 사양 및 제한

이 부록에서는 LVM 제품 사양에 대해 설명합니다.



참고: 하지만 이 제한에 맞게 구성한 시스템이라고 해서 자동으로 사용할 수 있는 것은 아닙니다.

표 A-1 볼륨 그룹 버전 최대값

	버전 1.0 볼륨 그룹	버전 2.0 볼륨 그룹	버전 2.1 볼륨 그룹
단일 HP-UX 시스템의 최대 데이터	128PB	1024PB	1024PB
시스템 상의 최대 볼륨 그룹 수	256	512 ¹	2048 ¹
볼륨 그룹에 있는 최대 물리 볼륨 수	255	511	2048
볼륨 그룹에 있는 최대 논리 볼륨 수	255	511	2047
물리 볼륨의 최대 크기	2TB	16TB	16TB
볼륨 그룹의 최대 크기	510TB	2048TB	2048TB
논리 볼륨의 최대 크기	16TB	256TB	256TB
물리 확장 영역의 최대 크기	256MB	256MB	256MB
스트라이프의 최대 크기	32MB	256MB	256MB
최대 스트라이프 수	255	511	511
논리 볼륨당 최대 논리 확장 영역 수	65535	33554432	33554432
물리 볼륨당 최대 물리 확장 영역 수	65535	16777216	16777216
최대 미러 사본 수 (MirrorDisk/UX 제품 필요)	2	5	5

1 2048 볼륨 그룹 제한은 버전 2.0과 버전 2.1 볼륨 그룹에 모두 적용됩니다. 두 버전의 볼륨 그룹은 0-2047개 볼륨 그룹을 사용하여 만들 수 있지만, 버전 2.0 볼륨 그룹의 최대 작성 개수는 512입니다.

표 A-2 버전 1.0 볼륨 그룹 제한

매개 변수	매개 변수 설정/변경 명령	최소 값	Default 값	최대 값
시스템 상의 볼륨 그룹 수	n/a	0	n/a	256
볼륨 그룹에 있는 물리 볼륨 수	vgcreate -p max_pv vgmodify -p max_pv	1	16	255
볼륨 그룹에 있는 논리 볼륨 수	vgcreate -l max_lv vgmodify -l max_lv	1	255	255
물리 볼륨 크기	pvcreate -s pv_size	1PE	LUN 용량	2TB
논리 볼륨 크기	lvcreate -L lv_size lvextend -L lv_size	0	0	16TB
물리 확장 영역 크기	vgcreate -s pe_size	1MB	4MB	256MB
스트라이프 크기	lvcreate -l stripe_size	4KB	8KB	32MB
스트라이프 수	lvcreate -i stripes	2	n/a	VG의 PV 수
논리 볼륨당 논리 확장 영역 수	lvcreate -l max_le lvextend -l max_le	0	0	65535
물리 볼륨당 물리 확장 영역 수	vgcreate -e max_pe vgmodify -e max_pe	1	1016 ¹	65535
미러 사본 수 (MirrorDisk/UX 제품 필요)	lvcreate -m copies lvextend -m copies	0	0	2

1 볼륨 그룹의 모든 첫 번째 물리 볼륨에 액세스하는 데 1016개 이상의 확장 영역이 필요할 경우 vgcreate 는 기본값을 물리 확장 영역 크기로 나눈 첫 번째 물리 볼륨의 크기로 증가시킵니다.

표 A-3 버전 2.x 볼륨 그룹 제한

매개 변수	매개 변수 설정/변경 명령	최소 값	Default 값	최대 값
시스템 상의 볼륨 그룹 수	n/a	0	n/a	2048 ¹
볼륨 그룹에 있는 물리 볼륨 수	n/a	511	511 (2.0) 2048 (2.1)	511 (2.0) 2048 (2.1)
볼륨 그룹에 있는 논리 볼륨 수	n/a	511	511 (2.0) 2047 (2.1)	511 (2.0) 2047 (2.1)
볼륨 그룹 크기	vgcreate -S max_vgsize	1MB ²	n/a	2 PB
물리 볼륨 크기	pvcreate -s pv_size	1PE	LUN 용량	16TB
논리 볼륨 크기	lvcreate -L lv_size lvextend -L lv_size	0	0	256TB
물리 확장 영역 크기	vgcreate -s pe_size	1MB	n/a	256MB
스트라이프 크기	lvcreate -l stripe_size	4KB	n/a	256MB
스트라이프 수	lvcreate -i stripes	2	n/a	VG의 PV 수
논리 볼륨당 논리 확장 영역 수	lvcreate -l max_le lvextend -l max_le	0	0	33554432
물리 볼륨당 물리 확장 영역 수	n/a	1	LUN 용량 ÷ PE 크기	16384
미러 사본 수 (MirrorDisk/UX 제품 필요)	lvcreate -m copies lvextend -m copies	0	0	5

1 2048 볼륨 그룹 제한은 버전 2.0과 버전 2.1 볼륨 그룹에 모두 적용됩니다. 두 버전의 볼륨 그룹은 0-2047개 볼륨 그룹을 사용하여 만들 수 있지만, 버전 2.0 볼륨 그룹의 최대 작성 개수는 512입니다.

2 지정된 물리 볼륨의 전체 크기가 max_vgsize보다 크면 vgcreate는 최소 볼륨 그룹 크기를 전체 크기로 조정합니다.

B LVM 명령 요약

이 부록에서는 LVM 명령과 사용에 대한 설명을 요약합니다.

표 B-1 LVM 명령 요약

명령	설명 및 예제
extendfs	파일 시스템을 확장합니다. # <code>extendfs /dev/vg00/rlvol3</code>
lvmdadm	볼륨 그룹 버전과 관련된 제한을 표시합니다. # <code>lvmdadm -t -V 2.0</code>
lvchange	논리 볼륨의 특성을 변경합니다. # <code>lvchange -t 60 /dev/vg00/lvol3</code>
lvcreate	볼륨 그룹에 논리 볼륨을 만듭니다. # <code>lvcreate -L 100 /dev/vg00</code>
lvdisplay	논리 볼륨 정보를 표시합니다. # <code>lvdisplay -v /dev/vg00/lvol1</code>
lvextend -m	논리 볼륨에 미러를 추가합니다. # <code>lvextend -m 1 /dev/vg00/lvol3</code>
lvextend -L	논리 볼륨 크기를 늘립니다. # <code>lvextend -L 120 /dev/vg00/lvol3</code>
lvlnboot	루트, 스왑 또는 덤프 영역이 될 논리 볼륨을 준비합니다. # <code>lvlnboot -d /dev/vg00/lvol2</code>
lvmerge	분리된 볼륨을 하나의 논리 볼륨으로 병합합니다. # <code>lvmerge /dev/vg00/lvol4b /dev/vg00/lvol4</code>
lvreduce -L	논리 볼륨의 크기를 줄입니다. # <code>lvreduce -L 100 /dev/vg00/lvol3</code>
lvreduce -m	논리 볼륨의 미러 사본 수를 줄입니다. # <code>lvreduce -m 0 /dev/vg00/lvol3</code>
lvremove	볼륨 그룹에서 논리 볼륨을 제거합니다. # <code>lvremove /dev/vg00/lvol6</code>
lvrmboot	루트, 스왑 또는 덤프에 대한 논리 볼륨 링크를 제거합니다. # <code>lvrmboot -d /dev/vg00/lvol2</code>

표 B-1 LVM 명령 요약 (계속)

명령	설명 및 예제
lvsplit	미러 논리 볼륨을 두 개의 논리 볼륨으로 분리합니다. # <code>lvsplit /dev/vg00/lvol4</code>
lvsync	못 쓰게 된 논리 볼륨 미러를 동기화합니다. # <code>lvsync /dev/vg00/lvol1</code>
pvchange	물리 볼륨의 특성을 변경합니다. # <code>pvchange -a n /dev/disk/disk2</code>
pvck	물리 볼륨에서 일관성 검사를 수행합니다. # <code>pvck /dev/disk/disk47_p2</code>
pvcreate	볼륨 그룹의 일부로 사용할 물리 볼륨을 만듭니다. # <code>pvcreate /dev/rdisk/disk2</code>
pvddisplay	물리 볼륨의 정보를 표시합니다. # <code>pvddisplay -v /dev/disk/disk2</code>
pvmove	한 물리 볼륨에서 다른 물리 볼륨으로 확장 영역을 이동합니다. # <code>pvmove /dev/disk/disk2 /dev/disk/disk3</code>
pvremove	물리 볼륨에서 LVM 데이터 구조를 제거합니다. # <code>pvremove /dev/rdisk/disk2</code>
vgcfgbackup	볼륨 그룹의 LVM 구성을 저장합니다. # <code>vgcfgbackup vg00</code>
vgcfgrestore	LVM 구성을 복원합니다. # <code>vgcfgrestore -n /dev/vg00 /dev/rdisk/disk2</code>
vgchange	볼륨 그룹을 설정하거나 해제합니다. # <code>vgchange -a y /dev/vg00</code>
vgchgid	물리 볼륨의 볼륨 그룹 ID를 변경합니다. # <code>vgchgid /dev/rdisk/disk3</code>
vgcreate	볼륨 그룹을 만듭니다. # <code>vgcreate /dev/vg01 /dev/disk/disk2 /dev/disk/disk3</code>
vgdisplay	볼륨 그룹에 대한 정보를 표시합니다. # <code>vgdisplay -v /dev/vg00</code>

표 B-1 LVM 명령 요약 (계속)

명령	설명 및 예제
vgextend	물리 볼륨을 추가하여 볼륨 그룹을 확장합니다. # <code>vgextend /dev/vg00 /dev/disk/disk2</code>
vgexport	시스템에서 볼륨 그룹을 제거합니다. # <code>vgexport /dev/vg01</code>
vgimport	기존 볼륨 그룹을 시스템에 추가합니다. # <code>vgimport -v /dev/vg04</code>
vgmodify	볼륨 그룹의 구성 매개 변수를 수정합니다. # <code>vgmodify -v -t -n -r vg32</code>
vgscan	시스템 디스크에서 볼륨 그룹을 검색합니다. # <code>vgscan -v</code>
vgreduce	볼륨 그룹에서 물리 볼륨을 하나 이상 제거하여 볼륨 그룹을 축소시킵니다. # <code>vgreduce /dev/vg00 /dev/disk/disk2</code>
vgremove	시스템과 디스크에서 볼륨 그룹의 정의를 제거합니다. # <code>vgremove /dev/vg00 /dev/disk/disk2</code>
vgsync	볼륨 그룹에서 미러 논리 볼륨을 모두 동기화합니다. # <code>vgsync vg00</code>

C 볼륨 그룹 프로비저닝 팁

이 부록에서는 볼륨 그룹을 만들 때 사용할 매개 변수에 대한 권장 사항을 제공합니다.

버전 1.0 볼륨 그룹의 최적 확장 영역 크기 선택

버전 1.0 볼륨 그룹을 만들 때 `vgcreate` 명령이 실행되지 않고 확장 영역 크기가 너무 작거나 해당 VGRA가 너무 크다는 메시지가 표시될 수 있습니다. 이 경우에는 더 큰 확장 영역 크기를 선택하고 `vgcreate`를 다시 실행해야 합니다.

미러 논리 볼륨에 쓰지 못하는 경우 확장 영역 크기를 늘리면 못 쓰게 된 데이터 영역이 늘어납니다. 따라서 못 쓰게 된 데이터를 재동기화하는 데 더 많은 시간이 소요될 수 있습니다. 또한 공간이 확장 영역 크기 단위로 할당되므로 각 논리 볼륨에 더 많은 공간이 할당될 수 있습니다. 따라서 원하는 구성 매개 변수를 적용하여 볼륨 그룹을 만드는 데 사용할 수 있는 최적 확장 영역 크기가 최소값이 됩니다.

볼륨 그룹의 최소 확장 영역 크기는 각 물리 볼륨당 물리 확장 영역 최대 개수(MAXPX)와 볼륨 그룹의 논리 볼륨 최대 개수(MAXLV) 및 물리 볼륨 최대 개수(MAXPV)를 사용하여 계산됩니다.

부팅 가능한 물리 볼륨이 있는 볼륨 그룹의 경우 메타데이터가 768KB 이내여야 합니다. 따라서 MAXLV 값과 함께 `vgcreate`를 실행할 경우 부팅 가능한 물리 볼륨이 없는 볼륨 그룹에서 지원되던 MAXPV 및 MAXPX가 부팅 가능한 물리 볼륨이 있는 볼륨 그룹에서 지원되지 않을 수 있습니다. 이러한 상황에서 부팅 가능한 물리 볼륨을 볼륨 그룹에 추가해야 하는 경우, 이러한 인수에 대해 더 작은 값을 지정하여 볼륨 그룹을 다시 만듭니다. 메타데이터 크기에서 가장 큰 요인은 분명히 MAXPV와 MAXPX의 값입니다. 또는 `-B` 옵션을 사용하지 않고 해당 물리 볼륨에서 `pvcreate`를 실행한 후 볼륨 그룹에 추가하여 부팅 가능한 물리 볼륨을 일반 물리 볼륨으로 변환합니다. 볼륨 그룹에 이미 포함되어 있는 물리 볼륨의 경우 `vgmodify`를 사용하여 부팅 가능한 물리 볼륨에서 일반 물리 볼륨으로 변경할 수 있습니다.

샘플 셸 스크립트

다음 셸 스크립트는 지정된 볼륨 그룹의 최소 확장 영역 크기를 표시하는 간단한 프로그램을 작성하여 컴파일합니다.

```

#!/usr/bin/sh
cat << EOF > vgrasize.c
#include <stdio.h>

#define BS 1024 /* Device block Size */
#define roundup(val, rnd) (((val + rnd - 1) / rnd) * rnd)

main(int argc, char *argv[])
{
    int i, length, lvs, pvs, pxs;
    if (argc != 4) {
        /* Usage example:
        * Maximum LVs in the VG = 255
        * Maximum PVs in the VG = 16
        * Maximum extents per PV = 2500
        *
        * $ vgrasize 255 16 2500
        */
        printf("USAGE: %s MAXLVs MAXPVs MAXPXs\n", argv[0]);
        exit(1);
    }
    lvs = atoi(argv[1]);
    pvs = atoi(argv[2]);
    pxs = atoi(argv[3]);
    length = 16 + 2 * roundup(2 +
        (roundup(36 + ((3 * roundup(pvs, 32)) / 8) +
            (roundup(pxs, 8) / 8) * pvs, BS) +
            roundup(16 * lvs, BS) +
            roundup(16 + 4 * pxs, BS) * pvs) / BS, 8);

    if (length > 768) {
        printf("Warning: A bootable PV cannot be added to a VG \n"
            "created with the specified argument values. \n"
            "The metadata size %d Kbytes, must be less \n"
            "than 768 Kbytes.\n"
            "If the intention is not to have a boot disk in this \n"
            "VG then do not use '-B' option during pvcreate(1M) \n"
            "for the PVs to be part of this VG. \n", length);
    }

    length = roundup(length, 1024) / 1024;

    if (length > 256) {
        printf("Cannot configure a VG with the maximum values"
            " for LVs, PVs and PXs\n");
        exit(1);
    }

    for (i = 1; i < length ; i = i << 1) { }

    printf("\nMinimum extent size for this configuration = %d MB\n", i);

    exit(0);
}
EOF
make vgrasize

```

용어

I/O 채널 분리	I/O 사용이 아주 많은 영역을 분리하는 경우에 유용한 디스크 구성. 예를 들어, 한 채널에 데이터베이스를 두고 다른 채널에 파일 시스템을 둘 수 있습니다. HP MirrorDisk/UX를 사용하여 논리 볼륨을 미러링하는 경우에는 여러 I/O 채널에 미러 사본을 분배하여 시스템과 데이터의 가용성을 높일 수 있습니다.
논리 볼륨	파일 시스템, 원시 데이터, 덤프 영역 또는 스왑을 저장할 수 있도록 크기가 유연한 가상 저장 장치. 데이터가 물리적이 아니라 논리적으로 분산되기 때문에 단일 논리 볼륨이 한 LVM 디스크에 매핑될 수도 있고 여러 디스크로 확장될 수도 있습니다. 논리 볼륨은 단일 디스크인 것처럼 관리자에게 표시됩니다.
논리 볼륨 관리자	가상(논리) 디스크를 구현하여 물리 디스크 액세스를 확장, 미러링하고 성능을 개선하는 운영 체제 소프트웨어 모듈
논리 확장 영역	논리 볼륨에서 크기가 고정되어 있고 주소를 지정할 수 있는 영역. 논리 볼륨의 기본 할당 단위인 논리 확장 영역은 물리 확장 영역에 매핑됩니다. 따라서 물리 확장 영역의 크기가 4MB이면 논리 확장 영역의 크기도 4MB가 됩니다. 논리 볼륨의 크기는 구성된 논리 확장 영역의 수로 결정됩니다.
동기화	논리 볼륨의 최신 사본에서 데이터를 복사하여 미러 논리 확장 영역의 오래된 사본을 업데이트하는 과정. 동기화를 사용하면 모든 사본에 같은 데이터가 포함되도록 하여 미러 논리 볼륨의 일관성을 유지할 수 있습니다.
디스크 스페닝	볼륨 크기가 단일 디스크의 크기를 초과할 수 있도록 여러 디스크에 걸쳐 논리 볼륨을 할당
물리 볼륨	LVM에서 볼륨 그룹에 포함할 수 있도록 초기화한 디스크. LVM 디스크라고도 합니다. 표준 디스크의 경우와 마찬가지로 LVM 디스크(물리 볼륨)에는 원시 장치 파일(예: <code>/dev/rdisk/disk3</code>)을 통해 액세스합니다. 디스크를 물리 볼륨으로 초기화하려면 HP SMH 또는 <code>pvccreate</code> 명령을 사용합니다.
물리 볼륨 그룹	물리 볼륨으로 구성된 볼륨 그룹 내의 하위 집합으로, 각각 별도의 I/O 채널 또는 인터페이스 어댑터를 사용하여 미러 데이터의 가용성을 높입니다.
물리 확장 영역	LVM 디스크에서 크기가 고정되어 있고 주소를 지정할 수 있는 영역. 물리 볼륨의 기본 할당 단위입니다. 물리 확장 영역은 논리 확장 영역이라는 논리 볼륨 상의 영역에 매핑됩니다.
미러링	데이터 가용성을 높이는 데이터 동시 복제. LVM에서는 동일한 논리 볼륨을 여러 LVM 디스크에 매핑하여 데이터 사본 하나가(다중 미러링의 경우 사본 여러 개) 손실되어도 쉽게 복구할 수 있는 수단을 마련합니다. 미러링을 사용하면 데이터 쓰기보다 많은 읽기를 사용하여 응용 프로그램의 데이터 액세스 속도를 높일 수 있습니다. 미러링에는 MirrorDisk/UX 제품이 필요합니다.
볼륨 그룹	개별 논리 볼륨에 디스크 공간을 할당할 수 있는 LVM 디스크 하나 이상의 모음. 한 디스크는 한 볼륨 그룹에만 속할 수 있습니다. 볼륨 그룹에는 볼륨 그룹의 디렉토리에 있는 그룹 파일(예: <code>/dev/vg01/group</code>)을 통해 액세스합니다. HP SMH 또는 <code>vgcreate</code> 명령을 사용하여 볼륨 그룹을 만듭니다.
쿼럼	볼륨 그룹을 변경 또는 활성화하기 위해 필요한 LVM 디스크 수 요구 사항. 볼륨 그룹을 활성화하려면 쿼럼에서 사용할 수 있는 LVM 디스크의 수가 볼륨 그룹을 마지막으로 활성화했을 때 구성되어 있던 LVM 디스크 수의 절반 이상이어야 합니다. 구성을 변경하기 위한 쿼럼 요구 사항은 절반 이상입니다. 쿼럼이 없는 경우에는 LVM에서 작업을 막습니다. 쿼럼은 구성 변경을 진행할 때(예: 논리 볼륨을 만들 때)와 상태 변경 시(예: 디스크가 실패한 경우)에 모두 확인합니다. 쿼럼을 사용하면 볼륨 그룹의 일관성과 무결성을 보장할 수 있습니다. <code>vgchange</code> 명령에 <code>-q n</code> 옵션을 사용하면 쿼럼 확인을 재정의할 수 있지만 이 경우 주의가 필요합니다.
할당 정책	디스크 공간이 논리 볼륨에 분산되는 방식과 LVM 디스크에 확장 영역을 배치하는 방법을 결정하는 LVM 할당 정책. LVM에서는 엄격/비엄격 및 연속/비연속으로 디스크 공간을 할당합니다. 엄격한 할당을 수행하려면 미러 사본이 다른 LVM 디스크에 있어야 합니다. 연속 할당을 수행하려면 단일 디스크 상에서 물리 확장 영역 사이에 간격이 없어야 합니다.

색인

심볼

/etc/default/fs, 106
/etc/fstab, 51, 68, 77, 106
/etc/lvmconf/ 디렉토리, 30, 51, 80
/etc/lvmpvg, 49
/etc/lvmtab, 22, 34, 54, 69, 82, 84, 116, 125, 139, 140
/stand/bootconf, 101, 104
/stand/rootconf, 117
/stand/system, 37
/var/adm/syslog/syslog.log, 117, 119, 123, 138, 149

B

BDRA
 lvlnboot로 업데이트, 51
 디스크 영역, 29
 부팅 디스크 요구 사항, 96
 손상, 124
Boot Data Reserved Area (참조 BDRA)

D

du 명령, 35

E

extendfs 명령, 35, 107, 157

F

fsadm 명령, 107, 108
fuser 명령, 67, 69, 79, 107, 109, 132
 및 NFS, 107, 109

H

HP SMH, 20, 53
 LVM 관리, 53
 명령 규칙, 25
 실행, 53

I

idisk 명령, 96, 102, 137
insf 명령, 96, 99, 102, 137
io_redirect_dsf 명령, 130, 133, 137
ioscan 명령, 51, 115, 130, 133, 137, 140
 인스턴스 번호 확인, 129, 133, 136
 장치 파일 확인, 59, 84
 하드웨어 경로 확인, 129, 131, 135

L

LIF 볼륨
 디스크 영역, 29
 부팅 디스크 요구 사항, 59, 96

유지 관리 모드 부팅, 116
Logical Interface Format (참조 LIF 볼륨)

lvadm 명령, 115, 139

lvchange 명령, 55, 65, 157
 동기화 정책 설정, 40, 99
 손상된 블록 배치, 120
 예약 정책 설정, 39
 오류, 140

 제한 시간 설정, 119

 할당 정책 설정, 38

lvcreate 명령, 55, 63, 157

 덤프 논리 볼륨용, 112
 동기화 정책 설정, 40
 스왑 논리 볼륨용, 111
 스트라이프 논리 볼륨, 64

 예약 정책 설정, 39

 크기 없음, 36

 할당 정책 설정, 38

lvdisplay 명령, 21, 41, 51, 55, 57, 115, 116, 157

 미러 상태 표시, 127

 예약 정책, 45

 제한 시간 값 표시, 119

 확장 영역 상태 표시, 127

lvextend 명령, 55

 논리 볼륨 확장, 65, 157
 미러 추가, 67, 100, 103, 157
 스왑 확장, 111
 오류, 104, 140
 특정 디스크로 확장, 66, 96
 파일 시스템 확장, 107

lvlnboot 명령, 23, 29, 51, 55, 157

 덤프 논리 볼륨용, 97, 112

 루트 논리 볼륨용, 97

 부팅 논리 볼륨용, 96, 97

 부팅 정보 업데이트, 51, 101, 104, 125

 부팅 정보 표시, 29, 97, 101, 104, 115, 127

 스왑 논리 볼륨용, 97, 111, 112

 오류, 142

LVM 정보 표시, 56

lvmdadm 명령, 24, 55, 122, 157

lvmdchk 명령, 54

lvmerge 명령, 55, 79, 157

 동기화, 41

lvreduce 명령, 55

 논리 볼륨 축소, 67, 157

 미러 제거, 68, 157

 및 pvmove 오류, 85

 스왑 장치 축소, 112

 특정 디스크에서 미러 제거, 68

 파일 시스템 축소, 108, 109, 125

lvremove 명령, 55, 69, 157

 볼륨 그룹 분리, 78

lvrmboot 명령, 23, 29, 55, 112, 125, 157

lvsplit 명령, 55, 79, 158

lvsync 명령, 41, 55, 158

M

minfree, 35

mkboot 명령, 29, 96, 100, 103, 138

mknod 명령, 60, 77, 83, 96, 159

N

newfs 명령, 48, 106, 109, 134

NFS

및 fuser, 107, 109

P

Physical Volume Reserved Area (참조 PVRA)

pvchange 명령, 54, 158

경로 비활성화, 94, 129, 132, 135

다중 경로 지정, 44

볼륨 그룹 구성 복원, 81

스페어링, 86, 87

오류, 142

제한 시간 설정, 119

pvck 명령, 23, 54, 116, 158

pvcreate 명령, 20, 23, 54, 59, 158

부팅 디스크, 59

부팅 디스크용, 96, 99, 103

스페어링, 86

pvdisplay 명령, 21, 51, 54, 57, 115, 116, 158

디스크를 사용하는 논리 볼륨 찾기, 127

스페어링, 43

pvlins (참조 다중 경로 지정)

pvmove 명령, 54, 84, 158

물리 확장 영역 0 이동, 71, 72

비정상적인 종료, 85

스페어링, 87

PVRA

디스크 영역, 29

pvremove 명령, 54, 158

S

scsimgr 명령

다중 경로 지정, 43, 94

디스크 교체 중, 129, 132, 136

System Management Homepage (참조 HP SMH)

V

vgcfgbackup 명령, 51, 54, 77, 80, 158

백업 위치, 80

오류, 140, 143

vgcfgrestore 명령, 54, 81, 115, 158

물리 볼륨이 부팅 가능한지 확인, 91, 92

백업 위치, 80

손상된 LVM 데이터 복구, 125

스페어 디스크 복귀, 86

오류, 143

vgchange 명령, 54, 99, 158

볼륨 그룹 비활성화, 69, 74, 76, 77, 82, 83, 89, 93

볼륨 그룹 정지, 76

볼륨 그룹 활성화, 70, 74, 77, 82, 83, 84, 90, 94, 122

분리된 모든 링크 연결, 95, 131, 134, 138

오류, 144

자중 동기화, 41

정지된 볼륨 그룹 다시 시작, 76

쿼럼 검사 비활성화, 78, 120

쿼럼 검사 활성화, 78, 121

활성화 장애, 120

vgchgid 명령, 55, 158

복제된 LUN 통합, 113

볼륨 그룹 분리, 77

vgcreate 명령, 54, 60, 61, 158

다중 경로 지정된 디스크 추가, 43

대체 부팅 디스크용, 96

오류, 147

vgdisplay 명령, 51, 54, 56, 115, 116, 158

사용 가능한 공간 표시, 63, 65, 141

스페어링, 43

오류, 147

정지된 동안, 75

vgdsf 명령, 55

vgexport 명령, 54, 69, 82, 159

디스크 이동, 83, 84

맵 파일, 69

볼륨 그룹 분리, 77

볼륨 그룹 이름 바꾸기, 76

vgextend 명령, 54, 62, 159

다중 경로 지정된 디스크 사용, 43, 70

스페어링, 86

오류, 148

VGID, 30, 145

vgexport 사용, 70

vgimport 사용, 70

변경, 113

볼륨 그룹 분리, 77

vgimport 명령, 54, 70, 82, 159

다중 경로 지정된 디스크 사용, 70

디스크 이동, 83, 84

볼륨 그룹 분리, 78

볼륨 그룹 이름 바꾸기, 77

오류, 149

vgmodify 명령, 23, 54, 70, 159

물리 볼륨 유형 변경, 91

물리 볼륨 크기 조정, 87, 141

볼륨 그룹 매개 변수 수정, 70, 149

정보 수집, 71

VGRA

디스크 영역, 30

및 vgmodify, 71

확장 영역 크기에 대한 크기 종속성, 30, 147

vgreduce 명령, 54, 62, 159

다중 경로 지정된 디스크 사용, 70

vgremove 명령, 54, 79, 159

vgscan 명령, 54, 159

/etc/lvmtab 다시 만들기, 125
디스크 이동, 83
vgsync 명령, 41, 54, 159
Volume Group Reserved Area (참조 VGRA)
VxFS 스냅샷 파일 시스템, 110

ㄱ

그룹 장치 파일, 60
기본 스왑 논리 볼륨, 37
 덤프 영역으로, 37
 미러링, 99

ㄴ

논리 볼륨, 37, 111
 (참조 덤프 논리 볼륨)
 (참조 루트 논리 볼륨)
 (참조 부팅 논리 볼륨)
 (참조 스왑 논리 볼륨)
 구성 정보, 82
 만들기, 63
 명령, 55
 명명 규칙, 27
 사용 중인 응용 프로그램 확인, 67, 69, 79, 107, 109, 132
 성능 문제, 34
 스왑용, 36
 이름 바꾸기, 68
 장치 파일, 27, 64, 68
 정보 표시, 57
 정의, 20
 제거, 69
 축소, 66
 크기, 33
 특정 디스크로 확장, 66
 특정 디스크에 만들기, 66
 확장, 65
논리 볼륨 만들기, 63
논리 볼륨 이름 바꾸기, 68
논리 볼륨 제거, 69
논리 볼륨 축소, 66
논리 볼륨 확장, 65
논리 볼륨에 미러 추가, 67
논리 볼륨에서 미러 제거, 68
 특정 디스크에서, 68
논리 볼륨을 특정 디스크로 확장, 66
논리 확장 영역
 물리 확장 영역에 매핑, 21
 및 논리 볼륨 크기 반올림, 35
 정의, 20
논리 확장 영역을 물리 확장 영역에 매핑, 21

ㄷ

다중 경로 지정, 43
 LVM 사용, 43
 경로 비활성화, 94
 다중 경로 지정된 디스크 이동, 83
 다중 경로 지정된 디스크 제거, 63

 링크 간 전환, 44
 볼륨 그룹 가져오기, 70
 설정, 43
 영구 장치 파일, 25
 원시, 25, 43
 이점, 43
 정의, 20
다중 경로 지정된 디스크 추가, 43
대체 링크 (참조 다중 경로 지정)
대체 부팅 디스크
 만들기, 95
대체 부팅 디스크 만들기, 95
덤프 논리 볼륨, 37
 만들기, 112
 요구 사항, 112
 제거, 112
 지침, 37
덤프 논리 볼륨 만들기, 112
덤프 논리 볼륨 제거, 112
데이터 복원
 볼륨 그룹 구성, 81
데이터 이동, 84
데이터베이스 파티션
 스트라이프 크기, 48
중기화 정책, 40
 미러 쓰기 캐시, 40
 미러 일관성 복구, 40
 없음, 40
디스크, 20
 (참조 물리 볼륨)
 이동, 82, 83
디스크 스트라이핑 (참조 스트라이핑)
디스크 스페어링 (참조 스페어링)
디스크 이동, 81, 84
디스크 장애, 126
디스크를 사용하는 논리 볼륨 찾기, 127

ㄹ

로그 파일, 117
루트 논리 볼륨, 97
 BDRA의 정보, 29
 lvlnboot, 97
 만들기, 96
 미러링, 99
 요구 사항, 95, 106
루트 볼륨 그룹
 및 덤프, 37

ㄴ

명명 규칙, 25
못 쓰게 된 데이터, 40
문제 해결 도구, 115
물리 볼륨
 경로 비활성화, 94
 데이터 이동, 84
 디스크 레이아웃, 28

- 만들기, 59
- 명령, 54
- 명명 규칙, 26
- 부팅 가능에서 부팅 불가능으로 변환, 91
- 이동, 82, 83
- 장치 파일, 26, 27, 138
- 정보 표시, 57
- 정의, 20
- 제거, 62
- 추가, 62
- 크기 조정, 87
- 물리 볼륨 그룹, 46, 48
 - 명명 규칙, 27
- 물리 볼륨 만들기, 59
- 물리 볼륨 이동, 82, 83
- 물리 볼륨 크기 조정, 87
- 물리 볼륨에 대한 경로 비활성화, 94
- 물리 볼륨을 부팅 가능에서 부팅 불가능으로 변환, 91
- 물리 확장 영역, 21
 - 다른 디스크로 이동, 63, 71
 - 동기화, 41
 - 물리 볼륨의 크기 초과, 91
 - 사용 중인 논리 볼륨 찾기, 127
 - 성능 고려 사항, 44
 - 쓰기 정책, 39
 - 정의, 20
 - 크기, 20, 30
 - 활당 정책, 38
- 미러 논리 볼륨, 38
 - I/O 채널 분리, 48
 - 기본 스왑 논리 볼륨, 99
 - 논리 볼륨 장치, 76
 - 논리 볼륨이 미러링되었는지 확인, 127
 - 논리 확장 영역을 물리 확장 영역에 매핑, 21
 - 동기화, 40, 158, 159
 - 동기화 정책, 40
 - 루트 논리 볼륨, 99
 - 못 쓰게 된 데이터, 40
 - 물리 논리 볼륨, 48
 - 물리 볼륨 그룹, 46
 - 미러 사본 만들기, 64
 - 미러 사본 수정, 64
 - 미러 쓰기 캐시, 45
 - 미러 제거, 68, 157
 - 미러 추가, 67, 157
- 백업, 79
- 병합, 79, 157
- 복구 가능한 오류 처리, 118
- 부팅 논리 볼륨, 99
- 부팅 디스크 교체, 135
- 부팅 디스크 미러링, 99
- 부팅 디스크 미러링,
 - HP 9000 서버, 99
 - HP Integrity 서버, 101
- 부팅할 수 없는 디스크 교체, 128
- 분리, 79, 158
- 스트라이핑, 48

- 스페어 디스크 만들기, 86
- 스페어 디스크 복귀, 86
- 엄격한 할당 정책, 141
- 예약 정책, 39, 45
- 오류 메시지, 140, 141, 150
- 유지 관리 모드, 117
- 이점, 38
- 정의, 19, 163
- 할당 정책, 38
- 미러 논리 볼륨 만들기, 64
- 미러 논리 볼륨 분리, 79
- 미러 논리 볼륨 수정, 64, 65
- 미러 논리 사본
 - 복구 불가능 오류 처리, 119
- 미러 동기화, 40
 - 수동, 41
 - 자동, 41
- 미러 볼륨 그룹
 - 스페어링, 42
- 미러 쓰기 캐시, 40
- 미러 일관성 복구, 40

N

백업

- VxFS 스냅샷 파일 시스템, 110
- 미러 논리 볼륨, 79
- 볼륨 그룹 구성, 80
- 버전 1.0 볼륨 그룹, 23, 28, 60, 153
- 버전 2.0 볼륨 그룹, 23, 153
- 버전 2.1 볼륨 그룹, 11, 23, 153
- 버전 2.x 볼륨 그룹, 11, 23, 28, 36, 37, 42, 55, 59, 61, 70, 86, 87, 91, 95, 99, 111, 112, 116, 122, 144, 146
- 병렬 예약 정책, 39
- 보조 번호, 28, 60
- 보조 스왑, 37
 - 구성, 111
- 볼륨 그룹
 - 가져오기, 70
 - 내보내기, 69
 - 만들기, 23, 60
 - 매개 변수 수정, 70
 - 명령, 55
 - 명명 규칙, 26
 - 분리, 77
 - 사용 가능한 공간, 63
 - 성능 고려 사항, 46
 - 이동, 81, 84
 - 이름 바꾸기, 76
 - 장치 파일, 27, 60, 69, 77, 83, 96
 - 정보 표시, 56
 - 정의, 31
 - 정지 및 다시 시작, 75
 - 제거, 78
 - 활성화 장애, 120, 123
- 볼륨 그룹 가져오기, 70
- 볼륨 그룹 구성
 - 백업, 80

- 복원, 81
- 위치, 80
- 볼륨 그룹 내보내기, 69
- 볼륨 그룹 만들기, 23, 60
- 볼륨 그룹 버전, 11
 - 버전 1.0, 23, 28, 60, 153
 - 버전 2.0, 23, 153
 - 버전 2.1, 23, 153
 - 버전 2.x, 23, 28, 36, 37, 42, 55, 59, 61, 70, 86, 87, 91, 95, 99, 111, 112, 116, 122, 144, 146
- 볼륨 그룹 분리, 77
- 볼륨 그룹 식별자 (참조 VGID)
- 볼륨 그룹 이름 바꾸기, 76
- 볼륨 그룹 정지, 75
- 볼륨 그룹 제거, 78
- 볼륨 그룹에 물리 볼륨 추가, 62
- 볼륨 그룹에서 물리 볼륨 제거, 62
- 볼륨 그룹의 매개 변수 수정, 70
- 부팅 논리 볼륨, 97
 - BDRA의 정보, 29
 - lvlnboot, 97
 - 미러링, 99
 - 요구 사항, 95
- 부팅 디스크 미러 만들기, 99
- 분리된 미러 논리 볼륨 병합, 79
- 블록 장치 파일, 26
- 비엄격 할당 정책, 39
- 비연속 할당, 23
 - 정의, 39

人

- 손상된 블록 배치, 95, 112, 119
- 순차 예약 정책, 39
- 스왑 논리 볼륨, 36, 37, 111
 - (참조 기본 스왑 논리 볼륨)
 - BDRA의 정보, 29
 - IODC 오류, 142
 - lvlnboot, 97, 111, 112
 - 만들기, 97, 111
 - 미러링 정책, 40, 99
 - 보조 스왑, 37
 - 성능 고려 사항, 36
 - 스트라이프 크기, 48
 - 요구 사항, 23, 95, 112
 - 유지 관리 모드, 117
 - 인터리빙, 36
 - 정보 인쇄, 29
 - 지침, 36
 - 크기 조정, 111
 - 확장, 111
- 스왑 논리 볼륨 만들기, 111
- 스왑 논리 볼륨 크기 조정, 111
- 스왑 논리 볼륨 확장, 111
- 스트라이프 논리 볼륨 만들기, 64
- 스트라이프 크기, 48
- 스트라이핑, 46
 - 및 미러링, 48

- 설정, 46
- 성능 고려 사항, 47
- 스트라이프 논리 볼륨 만들기, 64
- 스트라이프 크기 선택, 48
- 이점, 46
- 인터리브 디스크, 47
- 정의, 20
- 스패닝
 - 성능 고려 사항, 46
 - 정의, 19
- 스페어 디스크 만들기, 86
- 스페어 디스크 복귀, 86
- 스페어링, 23, 42
 - 링크 분리, 95
 - 스페어 디스크 만들기, 86
 - 스페어 디스크 복귀, 86
 - 요구 사항, 42
 - 정의, 19, 42

○

- 엄격한 할당 정책, 39
- 연속 할당
 - 덤프 논리 볼륨용, 112
 - 및 논리 볼륨 크기, 35
 - 스왑 논리 볼륨용, 111
 - 정의, 23, 39
- 예약 정책, 39
 - 병렬, 39
 - 순차, 39
- 오류 처리, 117
 - 미디어 오류, 119
 - 복구 가능한 오류, 118
 - 복구 불가능 오류, 119
- 원시 데이터 논리 볼륨
 - 스트라이프 크기, 48
- 유지 관리 모드 부팅, 99, 124, 125
 - 부팅, 116
 - 부팅 논리 볼륨 요구 사항, 96
- 이중 케이블(이중 컨트롤러) (참조 다중 경로 지칭)
- 인터리브 스와핑, 111

ㅈ

- 장애가 발생한 디스크 교체, 126
 - 미러, 128, 135
 - 미러링되지 않음, 131
- 장치 파일
 - 기존, 25, 26, 43, 94
 - 논리 볼륨, 27, 64, 68
 - 만들기, 60, 77, 83, 96, 99, 102
 - 물리 볼륨, 26, 27, 84, 99, 102, 138
 - 볼륨 그룹, 27, 60, 69, 77, 83, 96
 - 블록, 26
 - 영구, 25, 26
 - 특성, 26
 - 형식, 27
- 정지된 볼륨 그룹 다시 시작, 75

주 번호, 28, 60

ㄱ

쿼럼, 49

부팅 시 재정의, 100, 103, 121, 136

부팅 요구 사항, 124

오류 메시지, 121

재정의, 78, 120, 121, 163

정의, 120, 163

활성화, 78

ㄷ

특성 장치 파일, 26

ㅍ

파일 시스템 논리 볼륨, 34

/etc/fstab, 106

HFS 스트라이프 크기, 48

OnlineJFS, 106

루트 파일 시스템 (참조 루트 논리 볼륨)

만들기, 105

미러링을 통한 백업, 79

및 /etc/default/fs, 106

부팅 파일 시스템 (참조 부팅 논리 볼륨)

사용 중인 응용 프로그램 확인, 69, 79, 107, 109, 132

성능 고려 사항, 36

오버헤드, 35

응답하지 않음, 118

지침, 36

짧은 또는 긴 파일 이름, 106

초기 크기, 34

축소, 108, 125

HFS 또는 VxFS, 109

OnlineJFS, 108

크기 조정, 35

확장, 106

파일 시스템 논리 볼륨 만들기, 105

파일 시스템 논리 볼륨 확장, 106

파일 시스템 논리 볼륨의 크기 축소, 108

ㅎ

할당 정책, 38

엄격 및 비엄격, 39

연속 및 비연속, 39

핫 스왑 가능 디스크, 126