



Solaris 10 5/09 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

부품 번호: 820-7540-10
2009년 4월

Copyright 2009 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 모든 권리는 저작권자의 소유입니다.

Sun Microsystems, Inc.는 이 문서에 설명된 제품의 기술 관련 지적 재산권을 소유합니다. 특히 이러한 지적 재산권에는 하나 이상의 미국 특허 및 추가 특허 또는 미국 및 기타 국가에서 특허 출원 중인 응용 프로그램이 포함될 수 있습니다.

미국 정부의 권리 - 상용 소프트웨어. 정부 사용자는 Sun Microsystems, Inc. 표준 사용권 계약과 FAR의 해당 규정 및 추가 사항의 적용을 받습니다.

이 배포판에는 타사에서 개발한 자료가 포함되어 있을 수 있습니다.

본 제품의 일부는 Berkeley BSD 시스템일 수 있으며 University of California로부터 라이선스를 취득했습니다. UNIX는 미국 및 다른 국가에서 X/Open Company, Ltd.를 통해 독점적으로 사용권이 부여되는 등록 상표입니다.

Sun, Sun Microsystems, Sun 로고, Solaris 로고, Java Coffee Cup 로고, docs.sun.com, Java 및 Solaris는 미국 및 다른 국가에서 Sun Microsystems, Inc. 또는 Sun Microsystems, Inc. 회사의 상표 또는 등록 상표입니다. 모든 SPARC 상표는 사용 허가를 받았으며 미국 및 다른 국가에서 SPARC International, Inc.의 상표 또는 등록 상표입니다. SPARC 상표가 있는 제품은 Sun Microsystems, Inc.가 개발한 아키텍처와 기타 상표에 기초합니다.

OPEN LOOK 및 SunTM Graphical User Interface는 사용자와 라이선스를 위해 Sun Microsystems, Inc.이 개발했습니다. Sun은 컴퓨터 산업을 위한 비주얼 또는 그래픽 사용자 인터페이스의 연구와 개발에 관한 Xerox의 선구자적 노력을 인정합니다. Sun은 OPEN LOOK GUI를 구현하거나 Sun의 서면 라이선스 계약서를 준수하는 Sun의 라이선스를 포괄하는 Xerox Graphical User Interface에 대한 비배타적 라이선스를 Xerox로부터 취득하여 보유합니다.

이 발행물에서 다루는 제품과 수록된 정보는 미국 수출 관리법에 의해 규제되며 다른 국가의 수출 또는 수입 관리법의 적용을 받을 수도 있습니다. 이 제품과 정보를 직간접적으로 핵무기, 미사일 또는 생화학 무기에 사용하거나 핵과 관련하여 해상에서 사용하는 것은 엄격하게 금지합니다. 미국 수출 금지 국가 또는 금지된 개인과 특별히 지정된 국민 목록을 포함하여 미국 수출 금지 목록에 지정된 대상으로의 수출이나 재수출은 엄격하게 금지됩니다.

본 설명서는 “있는 그대로” 제공되며 상업성, 특정 목적에 대한 적합성 또는 비침해성에 대한 모든 묵시적 보증을 포함하여 모든 명시적 또는 묵시적 조건, 표현 및 보증에 대해 어떠한 책임도 지지 않습니다. 이러한 보증 부인은 법적으로 허용된 범위 내에서만 적용됩니다.

목차

머리말	11
제1부 Solaris Live Upgrade로 업그레이드	15
1 Solaris 설치 계획 정보 위치	17
계획 및 시스템 요구 사항 정보 위치	17
2 Solaris Live Upgrade(개요)	19
Solaris Live Upgrade 소개	19
Solaris Live Upgrade 프로세스	20
부트 환경 만들기	22
RAID-1 볼륨 파일 시스템이 있는 부트 환경 생성	27
부트 환경 업그레이드	34
부트 환경 활성화	37
원래 부트 환경으로 폴백	38
부트 환경 관리	40
3 Solaris Live Upgrade(계획)	41
Solaris Live Upgrade 요구 사항	41
Solaris Live Upgrade 시스템 요구 사항	41
Solaris Live Upgrade 설치	42
Solaris Live Upgrade 디스크 공간 요구 사항	44
RAID-1 볼륨(미러)를 만드는 경우의 Solaris Live Upgrade 요구 사항	45
패키지 또는 패치로 시스템 업그레이드	45
업그레이드 및 패치 제한 사항	46
lucreate 명령을 사용하여 파일 시스템을 만드는 지침	46
파일 시스템에 대한 슬라이스 선택 지침	47

루트(/) 파일 시스템에 대한 슬라이스 선택 지침	47
미러된 파일 시스템에 대한 슬라이스 선택 지침	48
스왑 볼륨에 대한 슬라이스 선택 지침	49
공유 가능 파일 시스템에 대한 슬라이스 선택 지침	50
새 부트 환경 내용 사용자 정의	51
부트 환경 사이의 파일 동기화	52
/etc/lu/synclist에 파일 추가	52
부트 환경 간의 강제 동기화	53
다중 부트 환경 부트	54
Solaris Live Upgrade 문자 사용자 인터페이스	55
4 Solaris Live Upgrade를 사용하여 부트 환경 만들기(작업)	57
작업 맵: Solaris Live Upgrade 설치 및 부트 환경 만들기	57
Solaris Live Upgrade 설치	58
▼ pkgadd 명령을 사용하여 Solaris Live Upgrade 설치	58
▼ Solaris 설치 프로그램을 사용하여 Solaris Live Upgrade 설치	59
Solaris Live Upgrade에 필요한 패치 설치	61
새 부트 환경 생성	62
▼ 처음으로 부트 환경 만들기	62
▼ 부트 환경 만들기 및 파일 시스템 병합	65
▼ 부트 환경 만들기 및 파일 시스템 분할	66
▼ 부트 환경 만들기 및 스왑 재구성	68
▼ 목록을 사용하여 부트 환경 만들기 및 스왑 재구성	70
▼ 부트 환경 만들기 및 공유 가능 파일 시스템 복사	72
▼ 다른 소스에서 부트 환경 만들기	73
▼ Solaris Flash 아카이브용 빈 부트 환경 만들기	75
▼ RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기	77
▼ 부트 환경 만들기 및 내용 사용자 정의	82
5 Solaris Live Upgrade를 통해 업그레이드(작업)	87
작업 맵: 부트 환경 업그레이드	87
부트 환경 업그레이드	88
업그레이드 지침	88
▼ 부트 환경에서 네트워크 설치 이미지 업그레이드	89
▼ 여러 CD를 사용하여 네트워크 설치 이미지 업그레이드	90

▼부트 환경에 네트워크 설치 이미지에 패키지 추가	92
▼부트 환경에서 네트워크 설치 이미지에 패치 추가	93
▼부트 환경에 설치된 패키지에 대한 정보 가져오기	95
JumpStart 프로필을 사용하여 업그레이드	95
부트 환경에 Solaris Flash 아카이브 설치	102
▼부트 환경에 Solaris Flash 아카이브 설치	103
▼프로필을 사용하여 Solaris Flash 아카이브 설치	104
▼프로필 키워드를 사용하여 Solaris Flash 아카이브 설치	106
부트 환경 활성화	107
부트 환경 활성화에 대한 요구 사항 및 제한	108
▼부트 환경 활성화	109
▼부트 환경 활성화 및 파일 동기화	110
x86: GRUB 메뉴를 사용하여 부트 환경 활성화	111
▼x86: GRUB 메뉴를 사용하여 부트 환경 활성화	112
6 오류 복구: 원래 부트 환경으로 폴백(작업)	115
SPARC: 원래 부트 환경으로 폴백	116
▼SPARC: 성공적인 새 부트 환경 활성화 시의 폴백	116
▼SPARC: 실패한 부트 환경 활성화에서의 폴백	116
▼SPARC: DVD, CD 또는 네트워크 설치 이미지를 사용하여 원래 부트 환경으로 폴백	117
x86: 원래 부트 환경으로 폴백	118
▼x86: GRUB 메뉴를 사용하여 성공적인 새 부트 환경 활성화 시 폴백	118
▼x86: GRUB 메뉴를 사용하여 실패한 부트 환경 활성화 시의 폴백	120
▼x86: GRUB 메뉴와 DVD 또는 CD를 사용하여 실패한 부트 환경 활성화에서의 폴백	122
7 Solaris Live Upgrade 부트 환경 유지 관리(작업)	125
Solaris Live Upgrade 유지 관리 개요	125
모든 부트 환경의 상태 표시	126
▼모든 부트 환경의 상태 표시	127
사전 구성된 부트 환경 업데이트	127
▼이전에 구성된 부트 환경 업데이트	128
예약된 만들기, 업그레이드 또는 복사 작업 취소	128
▼예약된 만들기, 업그레이드 또는 복사 작업 취소	129

부트 환경 비교	129
▼ 부트 환경 비교	129
비활성 부트 환경 삭제	130
▼ 비활성 부트 환경 삭제	130
활성 부트 환경의 이름 표시	131
▼ 활성 부트 환경의 이름 표시	131
부트 환경의 이름 변경	131
▼ 비활성 부트 환경의 이름 변경	132
부트 환경 이름과 관련된 설명 추가 또는 변경	133
▼ 텍스트를 사용하여 부트 환경 이름에 대한 설명 추가 또는 변경	133
▼ 파일을 사용하여 부트 환경 이름에 대한 설명 추가 또는 변경	134
▼ 텍스트 설명을 사용하여 부트 환경 이름 결정	134
▼ 파일의 설명을 사용하여 부트 환경 이름 결정	135
▼ 이름을 사용하여 부트 환경 설명 결정	135
부트 환경의 구성 보기	136
▼ 부트 환경의 구성 보기	136
8 비전역 영역이 설치된 시스템에서 Solaris OS 업그레이드	137
비전역 영역이 설치된 경우 Solaris Live Upgrade를 사용하여 업그레이드(개요)	137
Solaris Zones 및 Solaris Live Upgrade 이해	138
비전역 영역에서 Solaris Live Upgrade 사용을 위한 지침(계획)	142
비전역 영역이 별도의 파일 시스템에 있을 때 부트 환경 만들기	143
비전역 영역이 설치된 경우 부트 환경 만들기 및 업그레이드(작업)	144
▼ 시스템에 비전역 영역이 설치된 경우 Solaris Live Upgrade로 업그레이드(작업)	144
비전역 영역이 설치된 시스템 업그레이드(예)	149
시스템에 비전역 영역이 설치된 경우 Solaris Live Upgrade로 업그레이드	149
비전역 영역이 포함된 부트 환경 관리	150
▼ 부트 환경의 비전역 영역 파일 시스템의 구성 보기	151
▼ 비전역 영역이 설치된 시스템의 부트 환경 비교	151
비전역 영역이 포함된 시스템에서 lumount 명령 사용	152
9 Solaris Live Upgrade (예)	153
Solaris Live Upgrade를 사용한 업그레이드의 예	153
Solaris Live Upgrade 사용 준비	153
부트 환경 만들기	156

비활성 부트 환경 업그레이드	156
부트 환경의 부트 가능 여부 검사	156
비활성 부트 환경 활성화	156
(선택 사항) 소스 부트 환경으로 폴백	157
RAID-1 볼륨의 한쪽 분리 및 업그레이드 예(미러)	161
기존 볼륨에서 Solaris 볼륨 관리자 RAID-1 볼륨으로의 마이그레이션 예	165
빈 부트 환경 만들기 및 Solaris Flash 아카이브 설치 예	165
빈 부트 환경 만들기	166
새 부트 환경에 Solaris Flash 아카이브 설치	166
새 부트 환경 활성화	167
10 Solaris Live Upgrade(명령 참조)	169
Solaris Live Upgrade 명령줄 옵션	169
제2부 Solaris Live Upgrade를 사용하여 업그레이드 및 ZFS 루트 풀로 마이그레이션	171
11 Solaris Live Upgrade 및 ZFS(개요)	173
ZFS와 함께 Solaris Live Upgrade 사용 소개	174
UFS 파일 시스템에서 ZFS 루트 풀로 마이그레이션	174
UFS 루트(/) 파일 시스템에서 ZFS 루트 풀로 마이그레이션	174
Solaris Volume Manager 볼륨이 구성된 UFS 파일 시스템을 ZFS 루트 파일 시스템으로 마이그레이션	176
ZFS 루트 풀에서 새 부트 환경 만들기	177
동일한 루트 풀 내에 새 부트 환경 만들기	177
다른 루트 풀에 새 부트 환경 만들기	179
현재 실행 중인 시스템 이외의 소스에서 새 부트 환경 만들기	181
비전역 영역이 설치된 시스템에서 ZFS 부트 환경 만들기	182
추가 자원	182
12 ZFS용 Solaris Live Upgrade(계획)	183
Solaris Live Upgrade 사용 시 시스템 요구 사항 및 제한	183
추가 자원	186

13	ZFS 루트 풀의 부트 환경 만들기	187
	UFS 파일 시스템을 ZFS 파일 시스템으로 마이그레이션	187
	▼ UFS 파일 시스템에서 ZFS 파일 시스템으로 마이그레이션하는 방법	188
	동일한 ZFS 루트 풀 내에 부트 환경 만들기	194
	▼ 동일한 ZFS 루트 풀 내에서 ZFS 부트 환경을 만드는 방법	194
	새 루트 풀에 부트 환경 만들기	198
	▼ 새 ZFS 루트 풀에 부트 환경을 만드는 방법	198
	현재 실행 중인 시스템 이외의 소스에서 부트 환경 만들기	203
	ZFS 부트 환경으로 폴백	204
	추가 자원	204
14	비전역 영역이 설치된 ZFS용 Solaris Live Upgrade	207
	비전역 영역이 설치된 시스템에 ZFS 부트 환경 만들기(개요 및 계획)	207
	비전역 영역이 설치된 UFS 루트(/) 파일 시스템에서 ZFS 루트 풀로 마이그레이션(작업)	208
	▼ 비전역 영역이 설치된 시스템에서 UFS 파일 시스템을 ZFS 파일 시스템으로 마이그레이션하는 방법	208
	추가 자원	214
제3부	부록	215
A	문제 해결(작업)	217
	네트워크 설치 설정 문제	217
	시스템 부트 관련 문제	218
	매체에서 부트할 때 나타나는 오류 메시지	218
	매체에서 부트할 때 발생하는 일반적인 문제	219
	네트워크에서 부트할 때 나타나는 오류 메시지	220
	네트워크에서 부트할 때 발생하는 일반적인 문제	223
	Solaris OS의 초기 설치	224
	▼ x86: IDE 디스크의 잘못된 블록 확인	224
	Solaris OS 업그레이드	226
	업그레이드할 때 나타나는 오류 메시지	226
	업그레이드할 때 발생하는 일반적인 문제	228
	▼ 업그레이드 실패 후에 업그레이드 계속 진행	229
	x86: GRUB 사용 시 Solaris Live Upgrade 문제	229

▼ Veritas VxVm을 실행하는 Solaris Live Upgrade를 사용하여 업그레이드하는 중에 시스템 장애 발생	231
x86: 기존 Service 분할 영역이 없는 시스템에서 기본적으로 만들지 않는 Service 분할 영역	233
▼ 네트워크 설치 이미지 또는 Solaris Operating System DVD에서 소프트웨어 설치 ...	234
▼ Solaris Software - 1 CD 또는 네트워크 설치 이미지에서 설치	234
B 추가 SVR4 패키지 요구 사항(참조)	235
현재 OS 수정 방지	235
절대 경로 사용	235
pkgadd -R 명령 사용	236
\$PKG_INSTALL_ROOT 및 \$BASEDIR 사이의 차이점 개요	236
스크립트 작성 지침	237
디스크 없는 클라이언트 호환성 관리	237
패키지 검증	238
설치 또는 업그레이드 중 사용자 상호 작용 방지	239
영역의 패키지 매개 변수 설정	240
배경 정보	243
C 업그레이드할 때 패치 분석기 사용(작업)	245
Solaris 업데이트 릴리스로 업그레이드	245
▼ analyze_patches 스크립트 실행	246
▼ 패치 분석기 출력 검토	247
용어집	249
색인	263

머리말

본 설명서에서는 네트워크에 연결되거나 연결되지 않은 SPARC® 및 x86 구조 기반 시스템에서 Solaris™ 운영 체제(OS)를 설치 및 업그레이드하는 방법에 대해 설명합니다.

본 설명서에서는 시스템 하드웨어 또는 기타 주변 기기를 설정하는 방법에 대해서는 설명하지 않습니다.

주 - 이 Solaris 릴리스에서는 SPARC 및 x86 제품군 프로세서 구조 UltraSPARC®, SPARC64, AMD64, Pentium 및 Xeon EM64T 시스템을 지원합니다. 지원되는 시스템은 **Solaris OS: Hardware Compatibility Lists**(<http://www.sun.com/bigadmin/hcl>)를 참조하십시오. 이 설명서에서는 플랫폼 유형에 따른 구현 차이가 있는 경우 이에 대하여 설명합니다.

이 문서에서 사용되는 x86 관련 용어의 의미는 다음과 같습니다.

- "x86"은 64비트 및 32비트 x86 호환 제품의 큰 제품군을 의미합니다.
- "x64"는 AMD64 또는 EM64T 시스템에 대한 특정 64비트 정보를 나타냅니다.
- "32비트 x86"은 x86 기반 시스템에 대한 특정 32비트 정보를 나타냅니다.

지원되는 시스템은 **Solaris OS: Hardware Compatibility Lists**를 참조하십시오.

본 설명서의 대상

본 설명서는 Solaris OS 설치를 담당하는 시스템 관리자를 대상으로 합니다. 본 설명서는 다음 유형의 정보를 모두 제공합니다.

- 네트워크에 연결된 환경에서 복수 Solaris 시스템을 관리하는 기업 시스템 관리자를 위한 고급 Solaris 설치 정보
- Solaris 업그레이드를 가꿈 수행하는 시스템 관리자를 위한 기본 Solaris 설치 정보

관련 문서

표 P-1에서는 시스템 관리자용 설명서를 나열합니다.

표 P-1 Solaris를 설치하는 시스템 관리자입니까?

설명	정보
시스템 요구 사항 또는 상위 수준의 계획 정보가 필요합니까? 또는 Solaris ZFS™ 설치, 부트, Solaris 영역™ 분할 기술 또는 RAID-1 볼륨 만들기에 대한 고급 수준의 개요가 필요합니까?	Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획
DVD 또는 CD 매체에서 단일 시스템을 설치해야 합니까? Solaris 설치 프로그램이 설치 과정을 안내합니다.	Solaris 10 5/09 설치 설명서: 기본 설치
시스템 중단 시간 없이 시스템을 업그레이드하거나 패치해야 합니까? 업그레이드할 때 Solaris Live Upgrade를 사용하여 시스템 중단 시간을 줄이십시오.	Solaris 10 5/09 설치 설명서: Solaris Live Upgrade 및 업그레이드 계획
네트워크 또는 인터넷을 통해 보안 설치를 수행해야 합니까? 원격 클라이언트를 설치하려면 WAN 부트를 사용하십시오. 또는 네트워크 설치 이미지에서 네트워크를 통해 설치해야 합니까? Solaris 설치 프로그램이 설치 과정을 안내합니다.	Solaris 10 5/09 설치 설명서: 네트워크 기반 설치
여러 시스템에 Solaris를 설치해야 합니까? 설치를 자동화하려면 JumpStart™를 사용하십시오.	Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치
여러 시스템을 빠르게 설치하거나 패치해야 합니까? Solaris Flash™ 아카이브를 만들고 복제 시스템에 OS 복사본을 설치하려면 Solaris Flash 소프트웨어를 사용합니다.	Solaris 10 5/09 설치 설명서: Solaris Flash 아카이브(작성 및 설치)
시스템을 백업해야 합니까?	System Administration Guide: Devices and File Systems 의 23 장, “Backing Up and Restoring UFS File Systems (Overview)”
문제 해결 정보, 알려진 문제의 목록 또는 이 릴리스의 패치 목록이 필요합니까?	Solaris 릴리스 노트
시스템이 Solaris에서 작동하는지 검증해야 합니까?	SPARC: Solaris Sun 하드웨어 플랫폼 안내서
이 릴리스에서 추가, 제거 또는 변경된 패키지를 확인해야 합니까?	Solaris 패키지 목록
시스템 및 장치가 Solaris SPARC 및 x86 기반 시스템과 기타 타사 공급업체 제품에서 작동하는지 검증해야 합니까?	x86 플랫폼용 Solaris 하드웨어 호환성 목록

설명서, 지원 및 교육

Sun 웹 사이트에서는 다음 추가 자원에 대한 정보를 제공합니다.

- 설명서 (<http://www.sun.com/documentation/>)
- 지원 (<http://www.sun.com/support/>)
- 교육 (<http://www.sun.com/training/>)

Sun은 여러분의 의견을 환영합니다

Sun은 설명서의 내용 개선에 노력을 기울이고 있으며, 여러분의 의견과 제안을 환영합니다. 다음 사이트에 여러분의 의견을 제출하여 주십시오. 의견을 보내시려면 <http://docs.sun.com>에서 Feedback(피드백)을 누르십시오.

활자체 규약

다음 표는 이 책에서 사용되는 활자체 규약에 대해 설명합니다.

표 P-2 활자체 규약

활자체 또는 기호	의미	예제
AaBbCc123	명령 및 파일, 디렉토리 이름; 컴퓨터 화면에 출력되는 내용입니다.	.login 파일을 편집하십시오. 모든 파일 목록을 보려면 <code>ls -a</code> 명령을 사용하십시오. machine_name% you have mail.
AaBbCc123	사용자가 입력하는 내용으로 컴퓨터 화면의 출력 내용과 대조됩니다.	machine_name% su Password:
AaBbCc123	새로 나오는 용어, 강조 표시할 용어입니다. 명령줄 변수를 실제 이름이나 값으로 바꾸십시오.	<code>rm filename</code> 명령을 사용하여 파일을 제거합니다.
AaBbCc123	책 제목, 장, 절	사용자 설명서 의 6장을 읽으십시오. 캐시는 로컬로 저장된 복사본입니다. 파일을 저장하면 안 됩니다 . 주: 일부 강조된 항목은 온라인에서 굵은체로 나타납니다.

명령 예의 셸 프롬프트

다음 표에서는 C 셸, Bourne 셸 및 Korn 셸에 대한 기본 UNIX® 시스템 프롬프트 및 슈퍼유저 프롬프트를 보여 줍니다.

표 P-3 셸 프롬프트

셸	프롬프트
C 셸	machine_name%
슈퍼유저용 C 셸	machine_name#
Bourne 셸 및 Korn 셸	\$
슈퍼유저용 Bourne 셸 및 Korn 셸	#

1

Solaris Live Upgrade로 업그레이드

여기에서는 Solaris Live Upgrade를 사용하여 비활성 부트 환경을 만들고 업그레이드하는 데 대한 개요와 설명을 제공합니다. 그런 다음, 부트 환경을 현재 부트 환경이 되도록 전환할 수 있습니다. 이 부분은 UFS 루트(/) 파일 시스템이 포함된 시스템용으로 작성되었습니다. 하지만 많은 명령이 ZFS 파일 시스템에서도 사용될 수 있습니다.

Solaris 설치 계획 정보 위치

본 설명서에서는 Solaris Live Upgrade 프로그램을 사용하여 Solaris 운영 체제를 업그레이드하는 방법에 대한 정보를 제공합니다. 본 설명서에서는 Solaris Live Upgrade 사용에 대해 알아야 하는 모든 정보를 제공하지만 시작하기 전에 당사의 설치 설명서 모음 중 계획 설명서를 읽어 보면 도움이 될 것입니다. 다음 참조에서는 시스템을 업그레이드하기 전에 필요한 유용한 정보를 제공합니다.

계획 및 시스템 요구 사항 정보 위치

Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획에서는 시스템 요구 사항, 파일 시스템 계획 지침 및 업그레이드 계획 등과 같은 높은 수준의 계획 정보를 제공합니다. 다음 목록은 계획 설명서에 있는 장을 설명하고 해당 장에 대한 링크를 제공합니다.

계획 설명서의 장 설명	참조
이 장에서는 Solaris 설치 프로그램의 새로운 기능에 대해 설명합니다.	Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획 의 2 장, “Solaris 설치의 새로운 기능”
이 장에서는 Solaris OS를 설치 또는 업그레이드하기 전에 결정해야 하는 내용에 대한 정보를 제공합니다. 예를 들어, 네트워크 설치 이미지 또는 DVD 매체의 사용 시기 결정에 대한 정보 및 모든 Solaris 설치 프로그램에 대한 설명을 제공합니다.	Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획 의 3 장, “Solaris 설치 및 업그레이드(로드맵)”
이 장에서는 Solaris OS를 설치 또는 업그레이드하기 위한 시스템 요구 사항에 대하여 설명합니다. 디스크 공간과 기본 스왑 공간 할당을 위한 일반적인 지침도 제공됩니다. 업그레이드 제한 사항에 대해서도 설명합니다.	Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획 의 4 장, “시스템 요구 사항, 지침 및 업그레이드(계획)”

계획 설명서의 장 설명	참조
<p>이 장에는 시스템을 설치 또는 업그레이드하기 위해 필요한 모든 정보를 수집하는 데 도움이 되는 점검 목록이 있습니다. 이 정보는 예를 들어 대화식 설치를 수행하는 경우에 유용합니다. 대화식 설치를 수행하는 데 필요한 모든 정보를 점검 목록에 기록할 수 있습니다.</p>	<p>Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획의 5 장, “설치 또는 업그레이드 전에 정보 수집(계획 수립)”</p>
<p>이 장에서는 Solaris OS 설치 또는 업그레이드와 관련된 여러 기술에 대한 개요를 제공합니다. 이 기술과 관련된 지침 및 요구 사항에 대해서도 설명합니다. 이 장에는 ZFS 설치, 부트, Solaris 영역 분할 기술 및 설치 시 만들 수 있는 RAID-1 볼륨에 대한 정보가 포함됩니다.</p>	<p>Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획의 제II부, “ZFS, 부트, Solaris 영역 및 RAID-1 볼륨 관련 설치 이해”</p>

Solaris Live Upgrade(개요)

이 장에서는 Solaris Live Upgrade 프로세스에 대해 설명합니다.

주 - 이 책에서는 **슬라이스**라는 용어를 사용하지만 일부 Solaris 설명서와 프로그램에서는 슬라이스를 분할 영역이라고 부르기도 합니다.

Solaris Live Upgrade 소개

주 - 이 장에서는 UFS 파일 시스템용 Solaris Live Upgrade에 대해 설명합니다. UFS 파일 시스템을 ZFS 루트 폴로 마이그레이션하거나 ZFS 루트 폴을 만들고 설치하는 개괄적인 내용은 11 장, “[Solaris Live Upgrade 및 ZFS\(개요\)](#)”를 참조하십시오.

Solaris Live Upgrade는 시스템이 작동 중일 때 시스템을 업그레이드할 수 있는 방법을 제공합니다. 현재 부트 환경이 실행되는 동안 부트 환경을 복제하고 복제물을 업그레이드할 수 있습니다. 또는 업그레이드가 아니라 부트 환경에서 Solaris Flash 아카이브를 설치할 수 있습니다. 기존 시스템 구성은 아카이브의 업그레이드 또는 설치에 영향을 받지 않고 정상적으로 기능을 합니다. 준비가 되면 시스템을 다시 부트하여 새 부트 환경을 활성화할 수 있습니다. 오류가 발생하면 간단히 다시 부트하여 원래 부트 환경으로 쉽게 되돌릴 수 있습니다. 이 스위치는 테스트 및 평가 프로세스에서 필요로 하는 일반적인 중단 시간을 제거합니다.

Solaris Live Upgrade를 사용하여 현재 실행 중인 시스템에 영향을 주지 않고 부트 환경을 복제할 수 있습니다. 그런 다음 다음과 같은 작업을 수행할 수 있습니다.

- 시스템 업그레이드
- 현재 부트 환경의 디스크 구성을 새 부트 환경에서 다른 파일 시스템 유형, 크기 및 레이아웃으로 변경

- 서로 다른 이미지의 다양한 부트 환경 유지. 예를 들어, 한 부트 환경은 최신 패치를 포함하도록 만들고 다른 부트 환경은 업데이트 릴리스를 포함하도록 만들 수 있습니다.

Solaris Live Upgrade를 사용하려면 기본적인 시스템 관리에 대한 이해가 필요합니다. 파일 시스템 관리, 마운트, 부트 및 스왑 관리 등과 같은 시스템 관리 작업에 대한 배경 정보는 [System Administration Guide: Devices and File Systems](#)를 참조하십시오.

Solaris Live Upgrade 프로세스

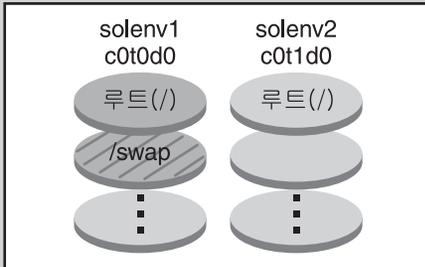
다음 개요에서는 현재 부트 환경의 복사본을 만들고 복사본을 업그레이드하며, 업그레이드된 복사본을 활성 부트 환경이 되도록 전환하는 데 필요한 작업에 대해 설명합니다. 또한 원본 부트 환경으로 전환하는 폴백 프로세스도 설명합니다.

[그림 2-1](#)에서는 전체 Solaris Live Upgrade 프로세스에 대해 설명합니다.

Solaris Live Upgrade 프로세스

① 부트 환경을 만듭니다.

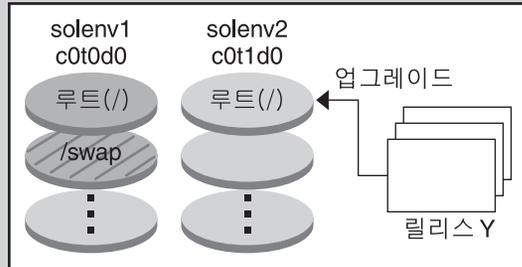
```
# lucreate -c solenv1 \
-m /:/dev/dsk/c0t1d0s0:ufs \
-n solenv2
```



② 비활성 부트 환경을 업그레이드합니다.

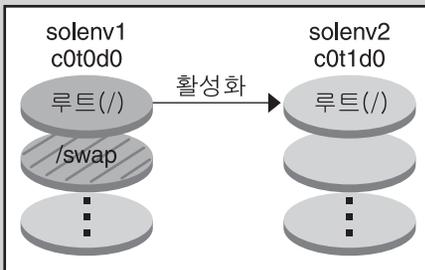
표준 업그레이드용:

```
(a) # luupgrade -u -n solenv2 \
-s /net/installmachine/export/Solaris/OS_image
```



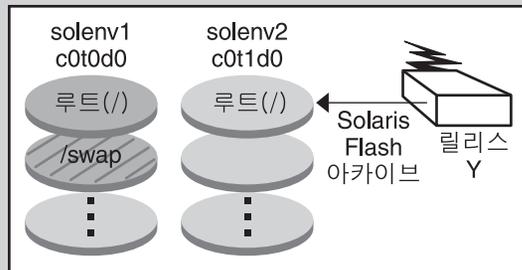
③ reboot 명령을 사용하여 비활성 부트 환경을 활성화합니다.

```
# luactivate solenv2
# init 6
```



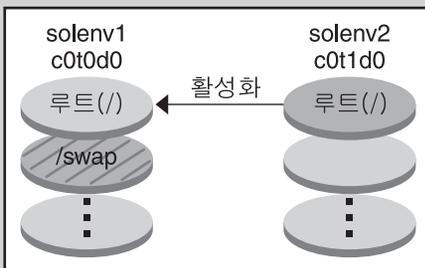
Solaris Flash 아카이브용:

```
(b) # luupgrade -f -n solenv2 \
-s /net/installmachine/export/Solaris/OS_image \
-a /net/server/archive/Solaris
```



④ (옵션) 원래 부트 환경으로 풀백합니다.

```
# luactivate solenv1
# init 6
```



⑤ (옵션) 비활성 부트 환경을 제거합니다.

```
# ludelete solenv2
```

그림 2-1 Solaris Live Upgrade 프로세스

다음 섹션은 Solaris Live Upgrade 프로세스를 설명합니다.

1. 새 부트 환경을 물리적 슬라이스 또는 논리 볼륨에 만들 수 있습니다.
 - 22 페이지 “부트 환경 만들기”
 - 27 페이지 “RAID-1 볼륨 파일 시스템이 있는 부트 환경 생성”
2. 34 페이지 “부트 환경 업그레이드”
3. 37 페이지 “부트 환경 활성화”
4. 38 페이지 “원래 부트 환경으로 폴백”

부트 환경 만들기

부트 환경을 만드는 프로세스는 중요 파일 시스템을 활성 부트 환경에서 새 부트 환경으로 복사하는 방법을 제공합니다. 필요할 경우 디스크가 재구성되며 파일 시스템이 사용자 지정되고 중요한 파일 시스템이 새 부트 환경으로 복사됩니다.

파일 시스템 유형

Solaris Live Upgrade는 중요 파일 시스템과 공유 가능 파일 시스템의 두 파일 시스템 유형을 구분합니다. 이들 파일 시스템 유형의 설명은 다음 표와 같습니다.

파일 시스템 유형	설명	예 및 자세한 내용
중요 파일 시스템	Solaris OS에서 필요한 중요 파일 시스템입니다. 이 파일 시스템은 활성 및 비활성 부트 환경의 <code>vfstab</code> 에 있는 별도의 마운트 지점입니다. 이 파일 시스템은 항상 소스에서 비활성 부트 환경으로 복사됩니다. 중요 파일 시스템은 때로 공유 불가 라고도 합니다.	예로는 <code>root(/)</code> , <code>/usr</code> , <code>/var</code> 또는 <code>/opt</code> 가 있습니다.
공유 가능한 파일 시스템	공유 가능 파일 시스템은 활성 및 비활성 부트 환경 모두의 <code>vfstab</code> 에 있는 동일한 마운트 지점을 포함하는 <code>/export</code> 와 같은 사용자 정의 파일들입니다. 따라서 활성 부트 환경에서 공유된 파일을 업데이트하면 비활성 부트 환경의 데이터 또한 업데이트됩니다. 새 부트 환경을 만들 때 공유 가능 파일 시스템은 기본적으로 공유됩니다. 하지만 대상 슬라이스를 지정할 수 있으며 그런 다음 파일 시스템이 복사됩니다.	<code>/export</code> 는 공유할 수 있는 파일 시스템의 예입니다. 공유 가능한 파일 시스템에 대한 자세한 내용은 50 페이지 “공유 가능 파일 시스템에 대한 슬라이스 선택 지침”을 참조하십시오.

파일 시스템 유형	설명	예 및 자세한 내용
스왑	<ul style="list-style-type: none"> UFS 파일 시스템의 경우 스왑은 공유 가능한 특수 볼륨입니다. 공유 가능 파일 시스템과 마찬가지로 모든 스왑 슬라이스는 기본적으로 공유됩니다. 하지만 스왑용 대상 디렉토리를 지정하면 스왑 슬라이스가 복사됩니다. ZFS 파일 시스템의 경우 스왑과 덤프 볼륨이 풀 내에서 공유됩니다. 	<ul style="list-style-type: none"> UFS 파일 시스템의 스왑 재구성에 대한 절차는 68 페이지 “부트 환경 만들기 및 스왑 재구성”를 참조하십시오. ZFS 루트 풀의 스왑에 대한 자세한 내용은 183 페이지 “Solaris Live Upgrade 사용 시 시스템 요구 사항 및 제한”를 참조하십시오.

파일 시스템에 RAID-1 볼륨 만들기

Solaris Live Upgrade는 파일 시스템에서 RAID-1 볼륨(미러)을 가진 부트 환경을 만들 수 있습니다. 개요는 [27 페이지](#) “RAID-1 볼륨 파일 시스템이 있는 부트 환경 생성”을 참조하십시오.

파일 시스템 복사

새 부트 환경을 만드는 프로세스는 중요한 파일 시스템을 복사할 수 있는 사용되지 않는 슬라이스를 식별하는 것부터 시작합니다. 슬라이스를 사용할 수 없거나 슬라이스가 최소 요건을 충족시키지 못하면 새 슬라이스를 포맷할 필요가 있습니다.

슬라이스를 정의한 뒤 파일 시스템을 디렉토리에 복사하기 전에 새 부트 환경에 파일 시스템을 재구성할 수 있습니다. `vfstab`을 편집하여 파일 시스템 디렉토리를 연결하거나 연결 해제하는 간단한 방법인 파일 시스템 분할과 병합을 통해 파일 시스템을 재구성합니다. 같은 마운트 지점을 지정하여 부모 디렉토리에 파일 시스템을 병합할 수 있습니다. 서로 다른 마운트 지점을 지정하여 부모 디렉토리에서 파일 시스템을 분리할 수도 있습니다.

비활성 부트 환경에 파일 시스템이 구성된 뒤 자동 복사를 시작합니다. 중요한 파일 시스템이 지정된 디렉토리로 복사됩니다. 공유 가능 파일 시스템은 복사되지 않지만 공유됩니다. 예외적으로 일부 공유 가능 파일 시스템을 복사하도록 지정할 수 있습니다. 활성 부트 환경의 파일 시스템을 비활성 부트 환경에 복사하면 파일은 새 디렉토리에 지정됩니다. 활성 부트 환경은 어떤 방법으로도 변경되지 않습니다.

파일 시스템 분할 또는 병합 절차

- [65 페이지](#) “부트 환경 만들기 및 파일 시스템 병합”
- [66 페이지](#) “부트 환경 만들기 및 파일 시스템 분할”

RAID-1 볼륨 파일 시스템이 있는 부트 환경 만들기 개요

[27 페이지](#) “RAID-1 볼륨 파일 시스템이 있는 부트 환경 생성”

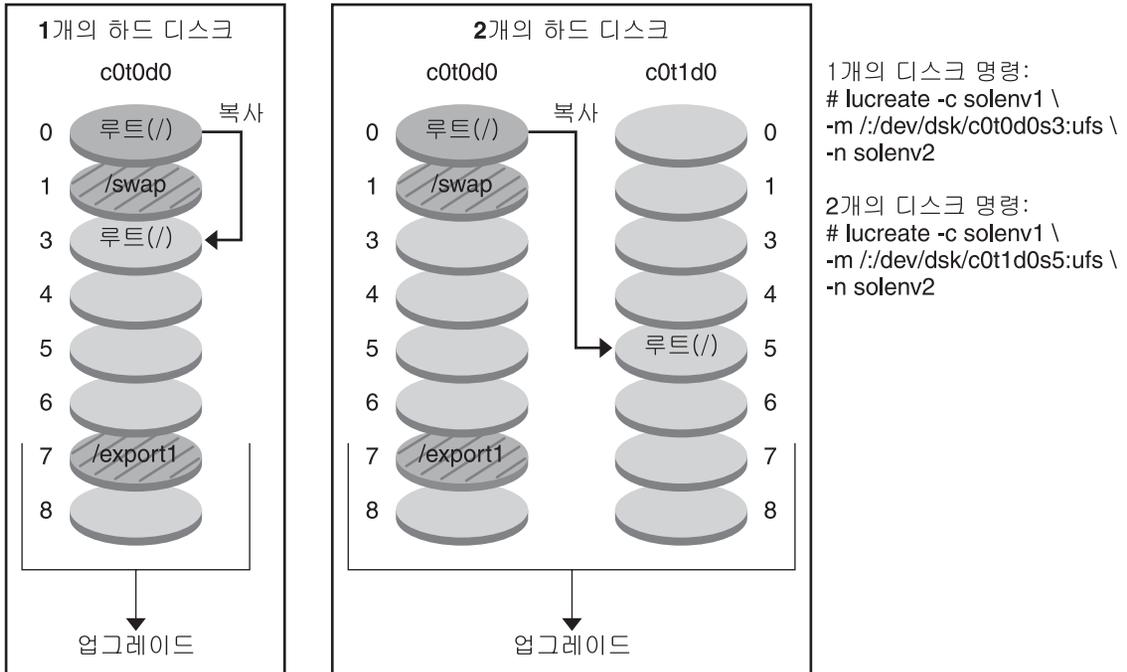
새 부트 환경 생성 예

UFS 파일 시스템의 경우 다음 그림은 새 부트 환경을 만드는 다양한 방법을 보여 줍니다.

ZFS 파일 시스템의 경우 11 장, “Solaris Live Upgrade 및 ZFS(개요)”를 참조하십시오.

그림 2-2에서는 새 부트 환경을 만들기 위해 중요 파일 시스템 루트(/)가 디스크의 다른 슬라이스로 복사되었음을 보여 줍니다. 활성 부트 환경은 하나의 슬라이스에 루트(/) 파일 시스템을 포함합니다. 새 부트 환경은 새 슬라이스에 루트(/) 파일 시스템이 있는 정확한 복제 부트 환경입니다. /swap 볼륨과 /export/home 파일 시스템은 활성 및 비활성 부트 환경에서 공유됩니다.

부트 환경 작성 -
루트(/) 파일 시스템을 단일 슬라이스에 복사



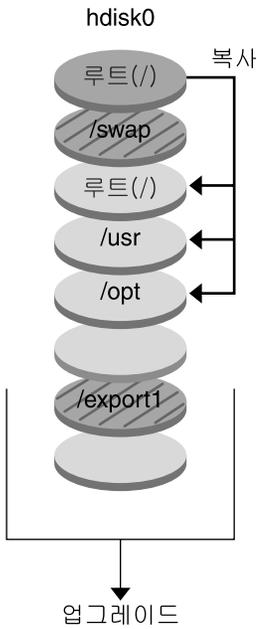
- 현재 릴리스 X
중요한 파일 시스템 루트(/)
- 비활성 릴리스 X
중요한 파일 시스템 루트(/)
- ▨ 공유 파일 시스템

그림 2-2 비활성 부트 환경 만들기 - 루트(/) 파일 시스템 복사

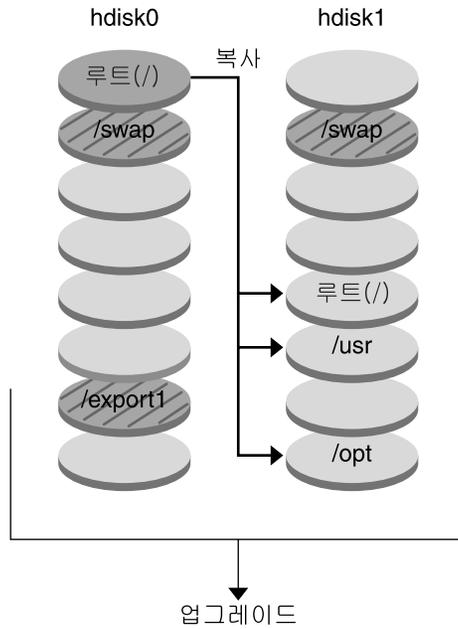
그림 2-3에서는 새 부트 환경을 만들기 위해 중요 파일 시스템이 분할되어 디스크의 슬라이스로 복사되었음을 보여 줍니다. 활성 부트 환경은 하나의 슬라이스에 루트(/) 파일 시스템을 포함합니다. 해당 슬라이스에서 루트(/) 파일 시스템은 /usr, /var 및 /opt 디렉토리를 포함합니다. 새 부트 환경에서는 루트(/) 파일 시스템이 분할되어 /usr 및 /opt가 별개의 슬라이스에 배치됩니다. /swap 볼륨과 /export/home 파일 시스템은 두 부트 환경 모두에서 공유됩니다.

부트 환경 작성 및 파일 시스템 분리

1개의 하드 디스크



2개의 하드 디스크

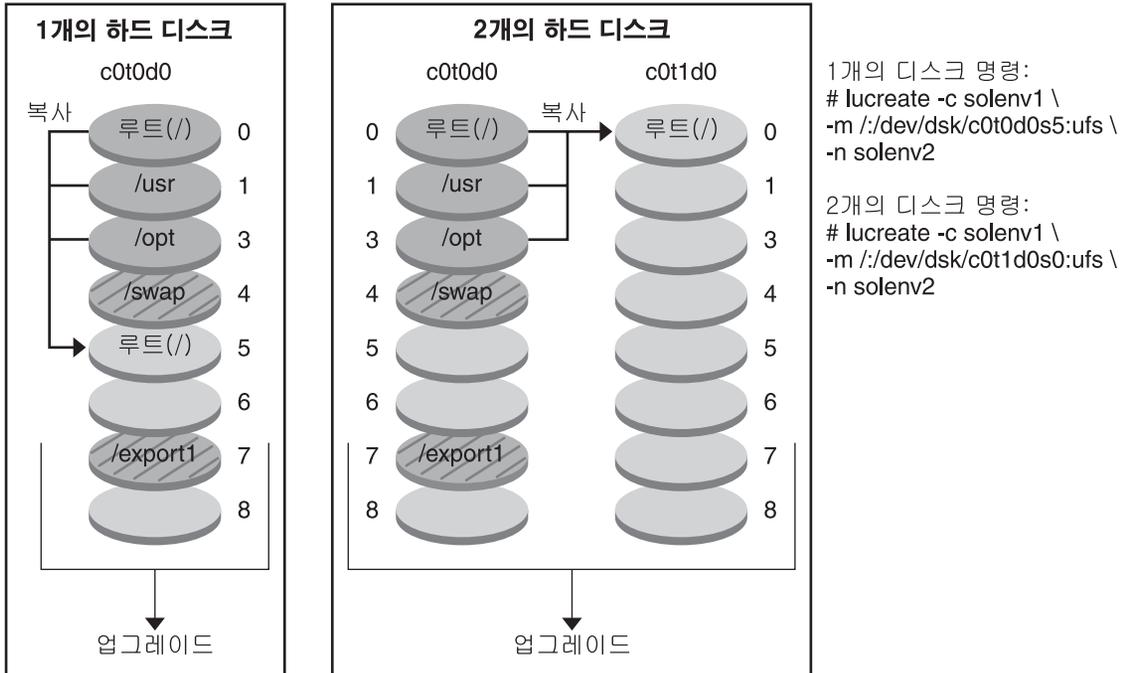


- 현재 릴리스 X
중요한 파일 시스템 루트(/)
- 비활성 릴리스 X
중요한 파일 시스템 루트(/) /usr /opt
- ▨ 공유 파일 시스템

그림 2-3 비활성 부트 환경 만들기 - 파일 시스템 분할

그림 2-4에서는 새 부트 환경을 만들기 위해 중요 파일 시스템이 병합되고 디스크의 슬라이스로 복사되었음을 보여 줍니다. 활성 부트 환경에는 루트(/) 파일 시스템, /usr, /var 및 /opt가 포함되어 있으며 자체 슬라이스에 각 파일 시스템이 있습니다. 새 부트 환경에서 /usr 및 /opt는 슬라이스의 루트(/) 파일 시스템에 병합됩니다. /swap 볼륨과 /export/home 파일 시스템은 두 부트 환경 모두에서 공유됩니다.

부트 환경 작성 - 파일 시스템 병합



- 현재 릴리스 X
중요한 파일 시스템 루트(/) /usr /opt
- 비활성 릴리스 Y
중요한 파일 시스템 루트(/)
- ▨ 공유 파일 시스템

그림 2-4 비활성 부트 환경 만들기 - 파일 시스템 병합

RAID-1 볼륨 파일 시스템이 있는 부트 환경 생성

Solaris Live Upgrade는 Solaris 볼륨 관리자 기술을 사용하여 RAID-1 볼륨에 캡슐화되는 파일 시스템이 포함될 수 있는 부트 환경을 만듭니다. Solaris Volume Manager는 볼륨을 사용하여 디스크를 안정적으로 관리할 수 있는 강력한 방법을 제공합니다. Solaris Volume Manager는 연결, 스트라이프 및 다른 복잡한 구성을 가능하게 합니다. Solaris Live Upgrade를 사용하면 루트(/) 파일 시스템에 대한 RAID-1 볼륨 만들기 등 해당 작업의 서브세트를 수행할 수 있습니다.

볼륨은 여러 디스크에 있는 디스크 슬라이스를 그룹화하여 OS에 있는 단일 디스크처럼 보이도록 합니다. Solaris Live Upgrade는 RAID-1 볼륨(미러) 내부에 단일 슬라이스

연결을 포함하는 루트(/) 파일 시스템에 대한 부트 환경만 만들 수 있습니다. 부트할 슬라이스를 하나만 선택하도록 부트 PROM이 제한되어 있기 때문입니다.

Solaris Live Upgrade 사용한 볼륨 관리

부트 환경을 만들 때 Solaris Live Upgrade를 사용하여 다음 작업을 관리할 수 있습니다.

- RAID-1 볼륨(미러)에서 단일 슬라이스 연결(하위미러)를 분리합니다. 필요한 경우 내용을 보존하여 새 부트 환경의 내용이 되도록 할 수 있습니다. 내용은 복사되지 않으므로 새 부트 환경을 빨리 만들 수 있습니다. 원래 미러에서 하위미러를 분리하면 하위미러는 더 이상 해당 미러에 속하지 않습니다. 하위미러에 대한 읽기 및 쓰기는 더 이상 미러를 통하여 수행되지 않습니다.
- 미러를 포함하는 부트 환경을 만듭니다.
- 최대 3개의 단일 슬라이스 연결을 새로 만든 미러에 연결합니다.

lucreate 명령을 -m 옵션과 함께 사용하여 미러를 만들고 하위미러를 분리하고 새 부트 환경에 하위미러를 연결할 수 있습니다.

주 - 현재 시스템에 VxVM 볼륨이 구성되어 있으면 lucreate 명령을 사용하여 새 부트 환경을 만들 수 있습니다. 데이터를 새 부트 환경으로 복사하면 Veritas 파일 시스템 구성이 손실되며 새 부트 환경에서 UFS 파일 시스템이 작성됩니다.

단계별 절차	77 페이지 “RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기”
설치 시 RAID-1 볼륨 만들기 개요	Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획 의 9장, “설치 중 RAID-1 볼륨(미러) 만들기(개요)”
Solaris Live Upgrade를 사용하는 경우 지원되지 않는 기타 복잡한 Solaris 볼륨 관리자 구성에 대한 자세한 내용	Solaris Volume Manager Administration Guide 의 2장, “Storage Management Concepts”

Solaris 볼륨 관리자 작업을 Solaris Live Upgrade로 매핑

Solaris Live Upgrade는 Solaris 볼륨 관리자 작업의 서브세트를 관리합니다. 표 2-1에서는 Solaris Live Upgrade가 관리할 수 있는 Solaris Volume Manager 구성 요소를 보여 줍니다.

표 2-1 볼륨 클래스

용어	설명
연결	RAID-0 볼륨입니다. 슬라이스가 연결되면 첫 번째 사용 가능한 슬라이스가 가득 찰 때까지 해당 슬라이스에 데이터가 기록됩니다. 슬라이스가 가득 차면 데이터는 연속적으로 다음 슬라이스에 기록됩니다. 연결은 미러에 포함된 경우를 제외하고 데이터 중복을 제공하지 않습니다.
미러	RAID-1 볼륨입니다. RAID-1 볼륨을 참조하십시오.
RAID-1 볼륨	여러 복사본을 유지하여 데이터를 복제하는 볼륨 클래스입니다. RAID-1 볼륨을 미러라고도 합니다. RAID-1 볼륨은 하위미러라고 하는 하나 이상의 RAID-0 볼륨으로 구성됩니다.
RAID-0 볼륨	스트라이프 또는 연결될 수 있는 볼륨 클래스입니다. 이 구성 요소는 하위미러라고도 합니다. 스트라이프 또는 연결은 미러의 기본 빌딩 블록입니다.
상태 데이터베이스	Solaris 볼륨 관리자 구성의 상태에 대한 디스크 정보를 저장하는 상태 데이터베이스입니다. 상태 데이터베이스는 복제된 여러 데이터베이스 복사본의 모음입니다. 각 복사본을 상태 데이터베이스 복제본이라고 합니다. 이 상태 데이터베이스는 모든 알려진 상태 데이터베이스 복제본의 위치 및 상태를 추적합니다.
상태 데이터베이스 복제본	상태 데이터베이스의 복사본입니다. 복제본은 데이터베이스에 있는 데이터가 유효한지 확인합니다.
하위 미러	RAID-0 볼륨을 참조하십시오.
볼륨	시스템에 하나의 논리적 장치로 보이는 물리적 슬라이스 또는 다른 볼륨의 그룹입니다. 볼륨은 응용 프로그램 또는 파일 시스템의 관점에서 보면 물리적 디스크와 기능적으로 동일합니다. 일부 명령줄 유틸리티에서는 볼륨을 메타 장치라고도 합니다.

Solaris Live Upgrade를 사용하여 RAID-1 볼륨 만들기 예

새 부트 환경용 RAID-1 볼륨을 만드는 명령 구문은 다음 예와 같습니다.

두 개의 물리적 디스크에 RAID-1 볼륨 만들기

그림 2-5에서는 두 개의 물리적 디스크에 생성되는 RAID-1 볼륨(미러)이 있는 새 부트 환경을 보여 줍니다. 다음 명령은 새 부트 환경과 미러를 만듭니다.

```
# lucreate -n second_disk -m /dev/md/dsk/d30:mirror,ufs \
-m /dev/dsk/c0t1d0s0,/dev/md/dsk/d31:attach -m /dev/dsk/c0t2d0s0,/dev/md/dsk/d32:attach \
-m -:/dev/dsk/c0t1d0s1:swap -m -:/dev/dsk/c0t2d0s1:swap
```

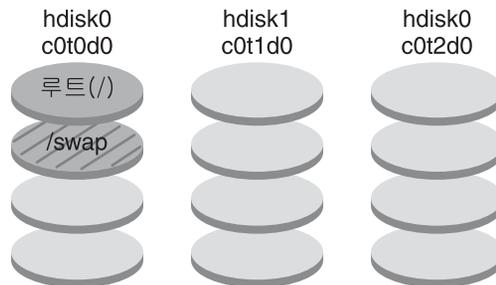
이 명령으로 다음 작업을 수행합니다.

- 새 부트 환경 second_disk를 만듭니다.
- 미러 d30을 만들고 UFS 파일 시스템을 구성합니다.

- 각 물리적 디스크의 슬라이스 0에 단일 장치 연결을 만듭니다. 연결 이름은 d31 및 d32로 지정됩니다.
- 두 연결을 미러 d30에 추가합니다.
- 루트(/) 파일 시스템을 미러에 복사합니다.
- 각 물리적 디스크의 슬라이스 1에 스왑을 위한 파일 시스템을 구성합니다.

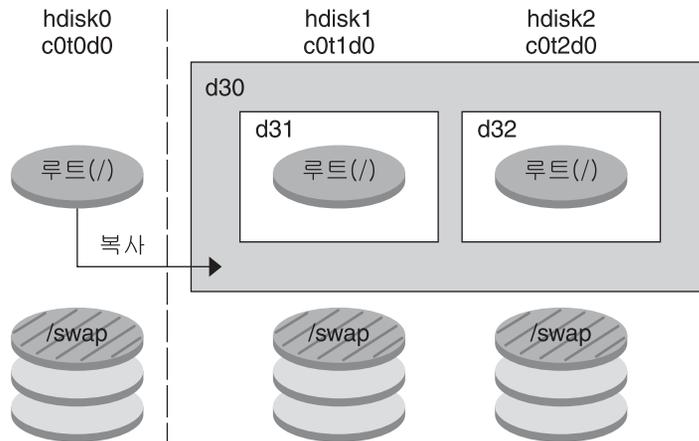
미러를 가진 새 부트 환경 작성

3개의 물리적 디스크가 있는 원본 시스템



```
명령: lucreate -n second_disk -m /:/dev/md/dsk/d30:mirror,ufs \
-m /:c0t1d0s0,d31:attach -m /:c0t2d0s0,d32:attach \
-m -:c0t1d0s1:swap -m -:c0t2d0s1:swap
```

새 부트 환경 second_disk



d30 – RAID-1 볼륨(미러)

d31 – 단일 슬라이스 연결(하위 미러)

d32 – 단일 슬라이스 연결(하위 미러)

그림 2-5 부트 환경 및 미러 만들기

부트 환경 만들기 및 기존 하위미러 사용

그림 2-6에서는 RAID-1 볼륨(미러)이 포함된 새 부트 환경을 보여 줍니다. 다음 명령은 새 부트 환경과 미러를 만듭니다.

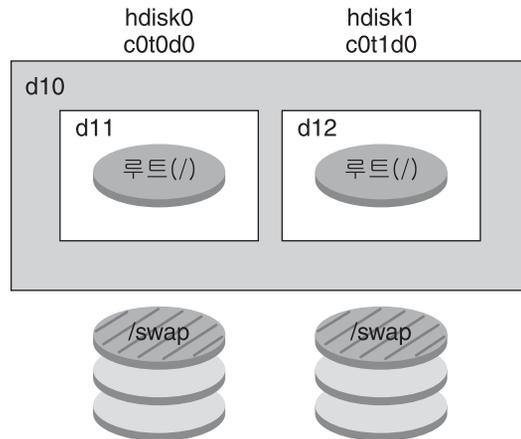
```
# lucreate -n second_disk -m /:/dev/md/dsk/d20:ufs,mirror \  
-m /:/dev/dsk/c0t1d0s0:detach,attach,preserve
```

이 명령으로 다음 작업을 수행합니다.

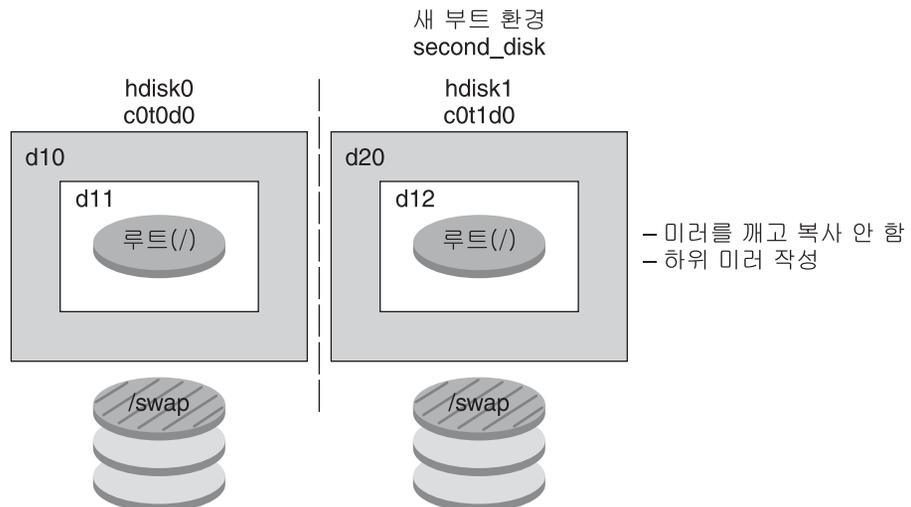
- 새 부트 환경 `second_disk`를 만듭니다.
- 미러 `d10`을 차단하고 연결 `d12`를 분리합니다.
- 연결 `d12`의 내용을 보존합니다. 파일 시스템은 복사되지 않습니다.
- 새 미러 `d20`을 만듭니다. 이제 두 개의 단방향 미러 `d10`과 `d20`이 만들어집니다.
- 연결 `d12`를 미러 `d20`에 연결합니다.

새 부트 환경 작성 및 기존의 하위 미러 사용

2개의 물리적 디스크가 있는 원본 시스템



```
명령: lucreate -n second_disk -m /:/dev/md/dsk/d20:ufs,mirror \
-m /:/dev/dsk/c0t1d0s0:detach,attach,preserve
```



d10 – RAID-1 볼륨(미러)
d11 – 단일 슬라이스 연결(하위 미러)
d12 – 단일 슬라이스 연결(하위 미러)
d20 – 새 RAID-1 볼륨(미러)

그림 2-6 부트 환경 만들기 및 기존 하위미러 사용

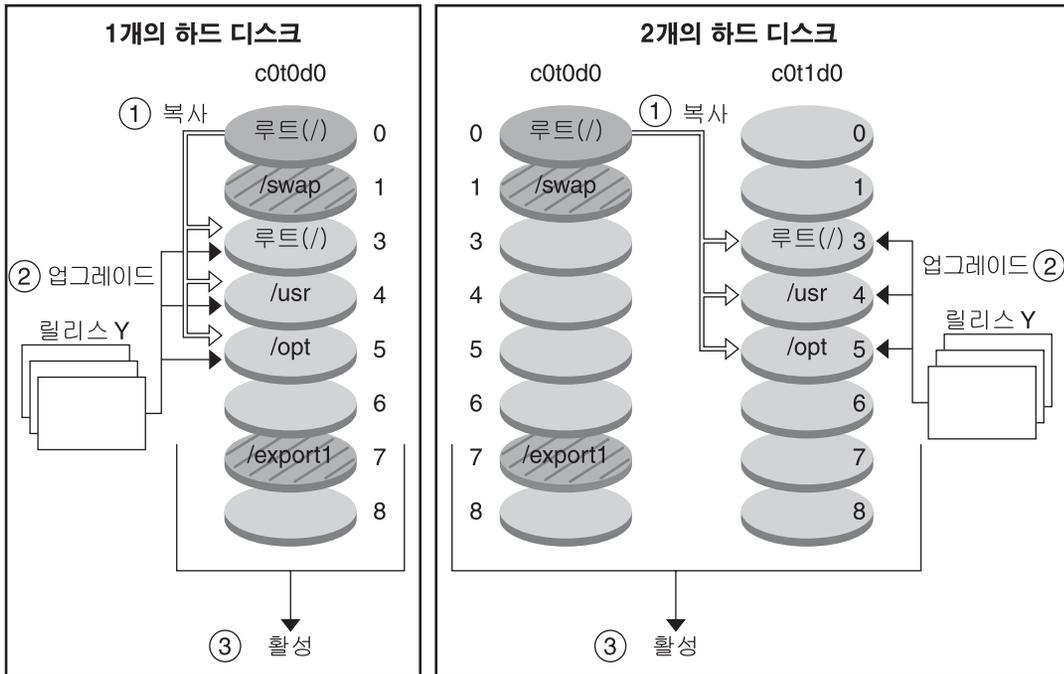
부트 환경 업그레이드

부트 환경을 만든 후에는 부트 환경에서 업그레이드를 수행할 수 있습니다. 해당 업그레이드 과정 중 부트 환경은 파일 시스템에 대해 RAID-1 볼륨(미러)을 포함할 수 있습니다. 또는 부트 환경에 비전역 영역이 설치되어 있을 수 있습니다. 업그레이드는 활성 부트 환경의 파일에는 전혀 영향을 주지 않습니다. 준비가 되면 새 부트 환경을 활성화하여 현재 부트 환경이 되도록 합니다.

UFS 파일 시스템용 부트 환경 업그레이드 절차	5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”
UFS 파일 시스템용 RAID-1 볼륨 파일 시스템이 있는 부트 환경 업그레이드 예	161 페이지 “RAID-1 볼륨의 한쪽 분리 및 업그레이드 예(미러)”
UFS 파일 시스템용 비전역 영역 업그레이드 절차	8 장, “비전역 영역이 설치된 시스템에서 Solaris OS 업그레이드”
ZFS 파일 시스템 업그레이드 또는 ZFS 파일 시스템으로 마이그레이션	11 장, “Solaris Live Upgrade 및 ZFS(개요)”

그림 2-7에서는 비활성 부트 환경으로 업그레이드합니다.

부트 환경 업그레이드



■ 현재 릴리스 X
중요한 파일 시스템 루트(/)

■ 비활성 릴리스 Y
중요한 파일 시스템 루트(/)/usr/opt

■ 공유 파일 시스템

① 단일 디스크 명령:

```
# lucreate -c solenv1 \
-m /:/dev/dsk/c0t0d0s3:ufs -m /usr:/dev/dsk/c0t0d0s4:ufs \
-m /opt:/dev/dsk/c0t0d0s5:ufs \
-n solenv2
```

① 2개의 디스크 명령:

```
# lucreate -c solenv1 \
-m /:/dev/dsk/c0t1d0s3:ufs -m /usr:/dev/dsk/c0t1d0s4:ufs \
-m /opt:/dev/dsk/c0t1d0s5:ufs \
-n solenv2
```

② # luupgrade -u -n solenv2 \

```
-s /net/installmachine/export/Solaris_10/OS_image
```

그림 2-7 비활성 부트 환경 업그레이드

또는 업그레이드가 아니라 부트 환경에서 Solaris Flash 아카이브를 설치할 수 있습니다. Solaris Flash 설치 기능을 사용하면 시스템의 Solaris OS에 대한 단일 참조 설치를 만들 수 있습니다. 이 시스템을 마스터 시스템이라고 합니다. 그런 다음 해당 설치를 복제 시스템이라 부르는 복수의 시스템에 복제할 수 있습니다. 이 상황에서 비활성 부트 환경은 복제 시스템입니다. 시스템에 Solaris Flash 아카이브를 설치할 때 아카이브는 초기 설치에서와 마찬가지로 기존 부트 환경의 모든 파일을 대체합니다.

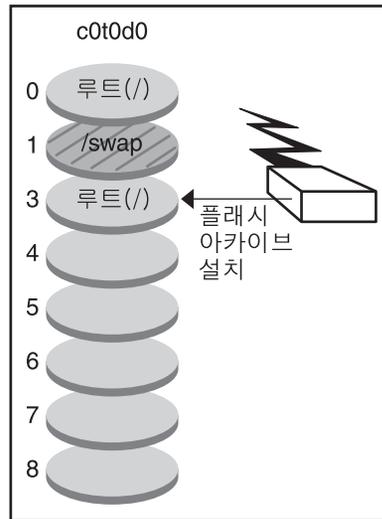
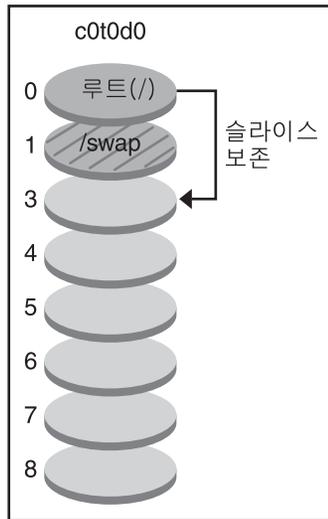
Solaris Flash 아카이브를 설치하는 절차는 102 페이지 “부트 환경에 Solaris Flash 아카이브 설치”를 참조하십시오.

다음 그림은 비활성 부트 환경에서 Solaris Flash 아카이브의 설치를 보여 줍니다. 그림 2-8에서는 단일 하드디스크가 있는 시스템을 보여 줍니다. 그림 2-9에서는 두 개의 하드 디스크가 있는 시스템을 보여 줍니다.

Solaris Flash 아카이브 설치 - 단일 디스크

① 빈 부트 환경을 만듭니다.

② 플래시 아카이브를 설치하여 업그레이드합니다.



- 현재 릴리스 X
핵심 파일 시스템 루트 (/)
- 비활성화 릴리스 Y
핵심 파일 시스템 루트 (/) /usr /opt
- 공유 파일 시스템

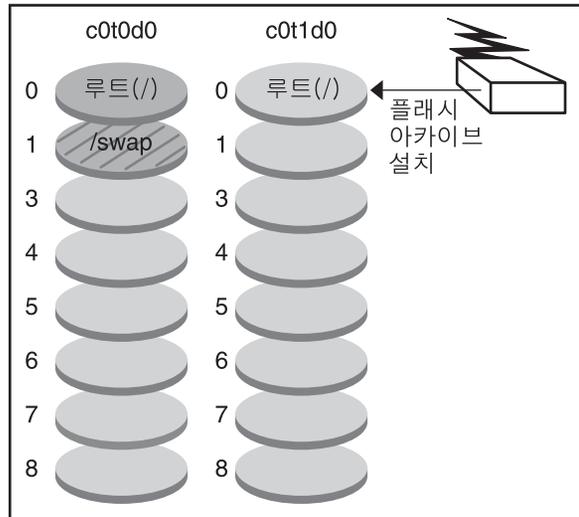
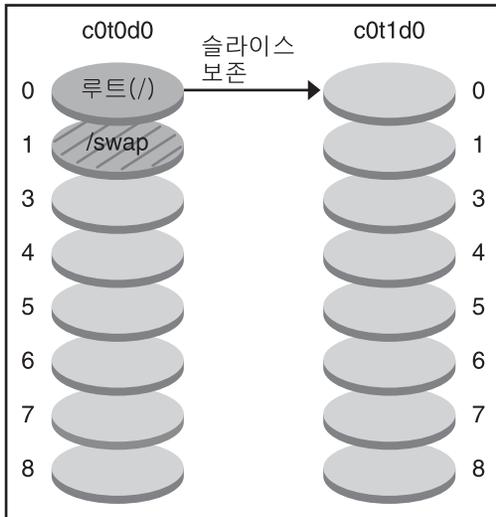
명령:
`# lucreate -s - \`
`-m /dev/dsk/c0t0d0s3:ufs -n solenv2`
`# luupgrade -f -n solenv2 \`
`-s /net/installmachine/export \`
`/Solaris/OS_image \`
`-a /net/server/archive/Solaris`

그림 2-8 단일 디스크에 Solaris Flash 아카이브 설치

Solaris Flash 아카이브 설치 - 두 개의 디스크

빈 부트 환경을 만듭니다.

플래시 아카이브를 설치하여 업그레이드합니다.



- 현재 릴리스 X
핵심 파일 시스템 루트(/)
- 비활성화 릴리스 X
핵심 파일 시스템 루트 (/)
- 공유 파일 시스템

```
명령:
# lucreate -s -\
-m /:/dev/dsk/c0t1d0s0 -n solenv2

# luupgrade -f -n solenv2 \
-s /net/installmachine/export \
/Solaris/OS_image \
-a /net/server/archive/Solaris
```

그림 2-9 두 개의 디스크에 Solaris Flash 아카이브 설치

부트 환경 활성화

새 부트 환경을 전환하여 활성화할 준비가 되면 새 부트 환경을 신속하게 활성화하고 재부트합니다. 새로 만든 부트 환경을 처음 부트할 때 부트 환경 간에 파일이 동기화됩니다. "동기화"란 특정 시스템 파일과 디렉토리가 가장 최신의 활성화 부트 환경에서 부트 중인 부트 환경으로 복사됨을 의미합니다. 시스템을 재부트했을 때, 새 부트 환경에 설치된 구성이 활성화됩니다. 따라서 원본 부트 환경은 비활성 부트 환경이 됩니다.

부트 환경을 활성화하는 절차

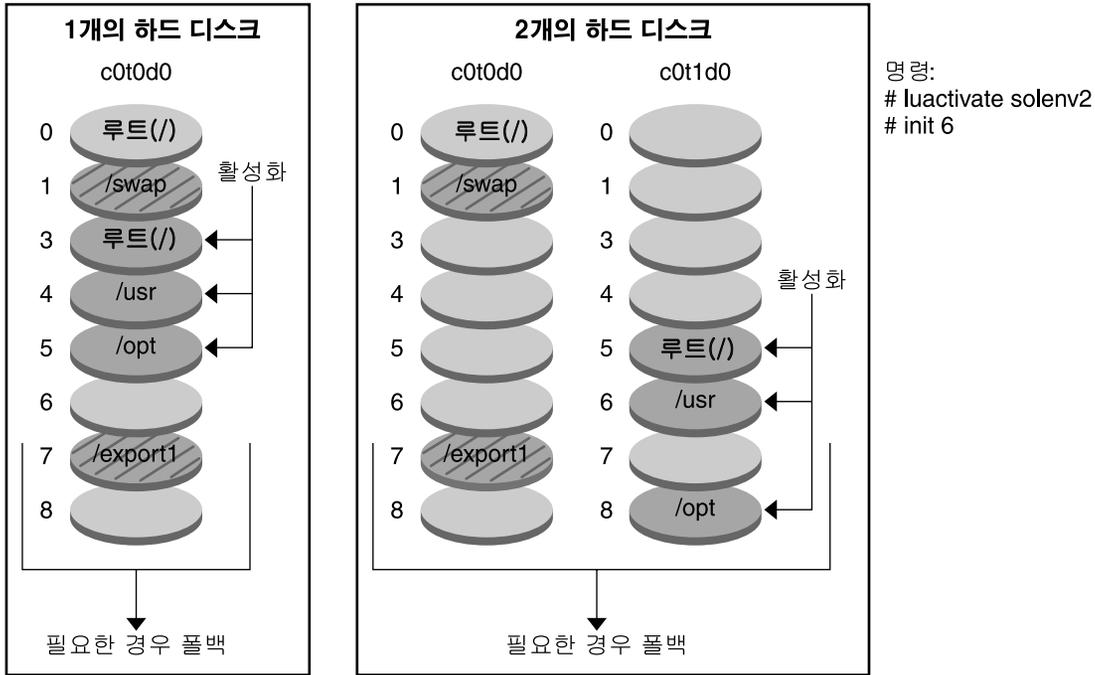
107 페이지 "부트 환경 활성화"

활성 및 비활성 부트 환경 동기화에 대한 내용

52 페이지 "부트 환경 사이의 파일 동기화"

그림 2-10에서는 비활성 부트 환경을 활성화 부트 환경으로 재부트한 후의 전환을 보여줍니다.

부트 환경 활성화



- 현재 릴리스 Y
중요한 파일 시스템 루트(/) /usr /opt
- 비활성 릴리스 X
중요한 파일 시스템 루트(/)
- ▨ 공유 파일 시스템

그림 2-10 비활성 부트 환경 활성화

원래 부트 환경으로 폴백

실패가 발생하면 활성화를 통해 원래 부트 환경으로 신속하게 폴백하고 재부트할 수 있습니다. 폴백 사용에는 시스템 재부트에 걸리는 시간 정도만이 소요되기 때문에 원래 부트 환경을 백업한 뒤 복원하는 것보다 훨씬 빠릅니다. 부트 실패한 새 부트 환경은 보존됩니다. 그런 다음 실패가 분석될 수 있습니다. luactivate에서 새 부트 환경을 활성화하는 데 사용한 부트 환경으로만 폴백할 수 있습니다.

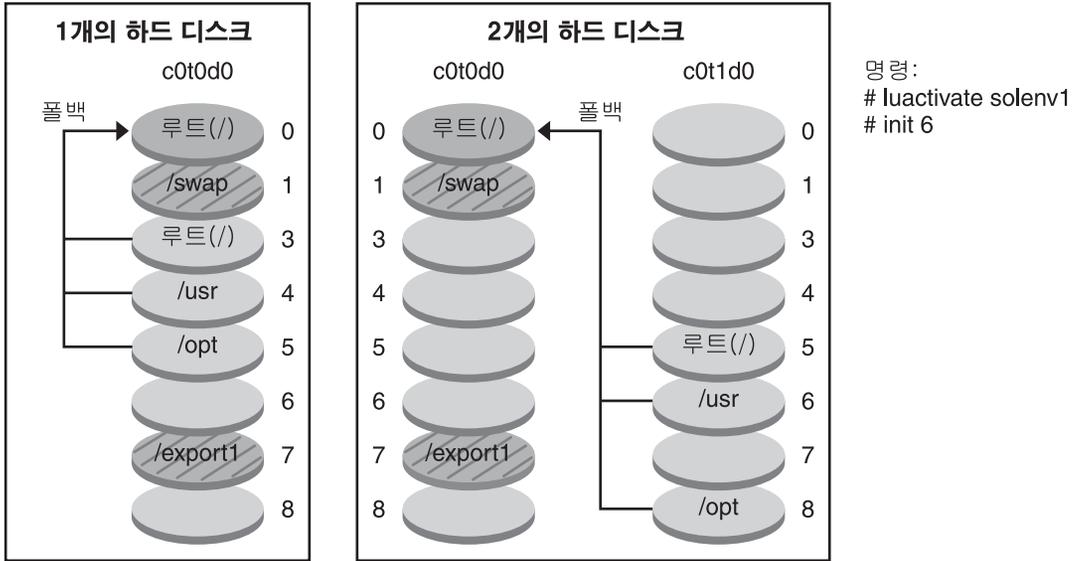
다음과 같은 방법으로 이전 부트 환경으로 폴백할 수 있습니다.

문제	작업
새 부트 환경이 성공적으로 부트되었지만 결과가 만족스럽지 않습니다.	<p>luactivate 명령과 이전 부트 환경 이름을 함께 실행하여 다시 부트합니다.</p> <p>x86 전용 - Solaris 10 1/06 릴리스부터 GRUB 메뉴에 있는 원래 부트 환경을 선택하여 폴백할 수 있습니다. 원래 부트 환경과 새 부트 환경은 GRUB 소프트웨어를 기반으로 해야 합니다. GRUB 메뉴에서 부트하면 이전 부트 환경과 새 부트 환경 간에 파일이 동기화되지 않습니다. 파일 동기화에 대한 자세한 내용은 53 페이지 “부트 환경 간의 강제 동기화”를 참조하십시오.</p>
새 부트 환경이 부트되지 않습니다.	<p>단일 사용자 모드에서 폴백 부트 환경으로 부트하고 luactivate 명령을 실행한 다음 재부트합니다.</p>
단일 사용자 모드에서 부트할 수 없습니다.	<p>다음 중 한 가지를 수행합니다.</p> <ul style="list-style-type: none"> ■ DVD나 CD 매체, 또는 네트워크 설치 이미지에서 부트합니다. ■ 폴백 부트 환경에 있는 루트(/) 파일 시스템을 마운트합니다. ■ luactivate 명령을 실행하고 다시 부트합니다.

폴백 절차는 6 장, “오류 복구: 원래 부트 환경으로 폴백(작업)”을 참조하십시오.

그림 2-11에서는 재부트하여 폴백할 때 수행되는 전환을 보여 줍니다.

원래 부트 환경으로의 폴백



- 현재 릴리스 X
중요한 파일 시스템 루트(/)
- 비활성 릴리스 X
중요한 파일 시스템 루트(/)
- 공유 파일 시스템

그림 2-11 원래 부트 환경으로의 폴백

부트 환경 관리

상태 검사, 부트 환경 이름 변경이나 삭제 등의 다양한 관리 활동을 수행할 수도 있습니다. 유지 관리 절차는 7장, “Solaris Live Upgrade 부트 환경 유지 관리(작업)”를 참조하십시오.

Solaris Live Upgrade(계획)

이 장에서는 Solaris Live Upgrade를 설치 및 사용하기 전에 검토할 지침 및 요구 사항을 제공합니다. 또한 업그레이드에 대한 일반 정보를 검토하려면 **Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획**의 “업그레이드 계획”을 참조하십시오.

주 - 이 장에서는 UFS 파일 시스템용 Solaris Live Upgrade에 대해 설명합니다. UFS 파일 시스템을 ZFS 루트 풀로 마이그레이션하거나 ZFS 루트 풀을 만들고 설치에 대한 계획 정보는 12 장, “ZFS용 Solaris Live Upgrade(계획)”를 참조하십시오.

이 장은 다음 절로 구성되어 있습니다.

- 41 페이지 “Solaris Live Upgrade 요구 사항”
- 45 페이지 “패키지 또는 패치로 시스템 업그레이드”
- 46 페이지 “lucreate 명령을 사용하여 파일 시스템을 만드는 지침”
- 47 페이지 “파일 시스템에 대한 슬라이스 선택 지침”
- 51 페이지 “새 부트 환경 내용 사용자 정의”
- 52 페이지 “부트 환경 사이의 파일 동기화”

Solaris Live Upgrade 요구 사항

Solaris Live Upgrade를 설치하고 사용하기 전에 다음 요구 사항을 숙지해야 합니다.

Solaris Live Upgrade 시스템 요구 사항

Solaris Live Upgrade는 Solaris 소프트웨어에 포함되어 있습니다. 현재 OS에 Solaris Live Upgrade 패키지를 설치해야 합니다. 반드시 Solaris Live Upgrade 패키지의 릴리스가 업그레이드하는 OS의 릴리스와 일치되어야 합니다. 예를 들어, 현재 OS가 Solaris 9 릴리스이며 Solaris 10 5/09 릴리스로 업그레이드하려는 경우 Solaris 10 5/09 릴리스의 Solaris Live Upgrade 패키지를 설치해야 합니다.

표 3-1에는 Solaris Live Upgrade에서 지원되는 릴리스가 나와 있습니다.

표 3-1 지원되는 Solaris 릴리스

현재 릴리스	호환되는 업그레이드 릴리스
Solaris 8 OS	Solaris 8, 9 또는 모든 Solaris 10 릴리스
Solaris 9 OS	Solaris 9 또는 모든 Solaris 10 릴리스
Solaris 10 OS	모든 Solaris 10 릴리스

Solaris Live Upgrade 설치

다음을 사용하여 Solaris Live Upgrade 패키지를 설치할 수 있습니다.

- pkgadd 명령. Solaris Live Upgrade 패키지는 SUNWLucfg, SUNWLur 및 SUNWLuui이며, 해당 패키지는 이 순서대로 설치해야 합니다.
- Solaris Operating System DVD, Solaris Software - 2 CD 또는 네트워크 설치 이미지의 설치 프로그램

Solaris Live Upgrade가 올바르게 작동하려면 다음 패치를 설치해야 합니다.

설명	자세한 정보
<p>주의: Solaris Live Upgrade가 올바르게 작동하려면 특정 OS 버전에 대한 제한된 패치 개정판 세트를 설치해야 합니다. Solaris Live Upgrade를 설치하거나 실행하기 전에 이러한 패치를 설치해야 합니다.</p> <p>x86 전용 - 이 패치 세트가 설치되어 있지 않으면 Solaris Live Upgrade가 실패하며 다음 오류 메시지가 표시될 수 있습니다. 다음 오류 메시지가 표시되지 않아도 필요한 패치가 아직 설치되어 있지 않을 수 있습니다. Solaris Live Upgrade를 설치하기 전에 항상 SunSolve info doc에 나열된 모든 패치가 설치되어 있는지 확인하십시오.</p> <pre>ERROR: Cannot find or is not executable: </sbin/biosdev>. ERROR: One or more patches required by Live Upgrade has not been installed.</pre> <p>정보 문서 206844(이전의 72099)에 나열된 패치는 언제든지 변경될 수 있습니다. 이러한 패치는 Solaris Live Upgrade의 결함과 Solaris Live Upgrade가 사용하는 구성 요소의 결함을 해결할 수 있습니다. Solaris Live Upgrade에서 문제가 발생하면 최신 Solaris Live Upgrade 패치가 설치되어 있는지 확인합니다.</p>	<p>http://sunsolve.sun.com을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 206844(이전의 72099)를 검색합니다.</p>

설명	자세한 정보
Solaris 8 또는 9 OS를 실행하는 경우 Solaris Live Upgrade 설치 프로그램을 실행하지 못할 수 있습니다. 이 릴리스에는 Java 2 런타임 환경을 실행하는 데 필요한 패치 집합이 포함되어 있지 않습니다. Solaris Live Upgrade 설치 프로그램을 실행하고 패키지를 설치하려면 반드시 Java 2 런타임 환경으로 권장되는 패치가 있어야 합니다.	Solaris Live Upgrade 패키지를 설치하려면 pkgadd 명령을 사용합니다. 또는 권장되는 패치 클러스터를 Java 2 런타임 환경용으로 설치합니다. 패치 클러스터는 http://sunsolve.sun.com 에 있습니다.

Solaris Live Upgrade 소프트웨어 설치에 대한 설명은 58 페이지 “Solaris Live Upgrade 설치”를 참조하십시오.

필수 패키지

Solaris Live Upgrade에 문제가 발생하면 패키지가 없는 경우일 수 있습니다. 다음 표에 나열된 패키지 중에서 Solaris Live Upgrade를 사용하는 데 필요한 패키지가 OS에 있는지 확인하십시오.

Solaris 10 릴리스의 경우:

- 다음 소프트웨어 그룹 중 하나를 설치하면 이러한 소프트웨어 그룹에 필요한 모든 Solaris Live Upgrade 패키지가 들어 있습니다.
 - 전체 Solaris 소프트웨어 그룹 및 OEM 지원
 - 전체 Solaris 소프트웨어 그룹
 - 개발자 Solaris 소프트웨어 그룹
 - 최종 사용자 Solaris 소프트웨어 그룹
- 다음 소프트웨어 그룹 중 하나를 설치하면 Solaris Live Upgrade를 사용하는 데 필요한 모든 패키지가 없을 수 있습니다.
 - 코어 시스템 지원 소프트웨어 그룹
 - 축소된 네트워크 지원 소프트웨어 그룹

소프트웨어 그룹에 대한 자세한 내용은 [Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획](#)의 “소프트웨어 그룹의 권장 디스크 공간”을 참조하십시오.

표 3-2 Solaris Live Upgrade에 대한 필수 패키지

Solaris 8 릴리스	Solaris 9 릴리스	Solaris 10 릴리스
SUNWadmap	SUNWadmap	SUNWadmap
SUNWadmc	SUNWadmc	SUNWadmlib-sysid
SUNWlibC	SUNWadmfw	SUNWadmr
SUNWbzip	SUNWlibC	SUNWlibC

표 3-2 Solaris Live Upgrade에 대한 필수 패키지 (계속)

Solaris 8 릴리스	Solaris 9 릴리스	Solaris 10 릴리스
SUNWgzip	SUNWgzip	Solaris 10 3/05에만 해당: SUNWgzip
SUNWj2rt	SUNWj2rt	SUNWj5rt
주-SUNWj2rt 패키지는 다음 조건에서만 필요합니다.	주-SUNWj2rt 패키지는 다음 조건에서만 필요합니다.	주-SUNWj5rt 패키지는 다음 조건에서만 필요합니다.
<ul style="list-style-type: none"> ■ Solaris Live Upgrade 설치 프로그램을 실행하여 Solaris Live Upgrade 패키지를 추가하는 경우 ■ CD 매체를 업그레이드 및 사용하는 경우 	<ul style="list-style-type: none"> ■ Solaris Live Upgrade 설치 프로그램을 실행하여 Solaris Live Upgrade 패키지를 추가하는 경우 ■ CD 매체를 업그레이드 및 사용하는 경우 	<ul style="list-style-type: none"> ■ Solaris Live Upgrade 설치 프로그램을 실행하여 Solaris Live Upgrade 패키지를 추가하는 경우 ■ CD 매체를 업그레이드 및 사용하는 경우

시스템의 패키지를 확인하려면 다음 명령을 입력합니다.

```
% pkginfo package_name
```

Solaris Live Upgrade 디스크 공간 요구 사항

업그레이드에 대한 일반 디스크 공간 요구 사항을 따르십시오. **Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획의 4장**, “시스템 요구 사항, 지침 및 업그레이드(계획)”를 참조하십시오.

부트 환경 만들기에 필요한 파일 시스템 크기를 예측하려면 새 부트 환경의 만들기를 시작합니다. 크기가 계산됩니다. 그런 다음 프로세스를 중지할 수 있습니다.

새 부트 환경의 디스크는 부트 장치로 서브할 수 있어야 합니다. 일부 시스템은 부트 장치로 서브할 수 있는 디스크를 제한합니다. 부트 제한이 적용되는지 여부를 결정하려면 시스템 설명서를 참조하십시오.

새 부트 환경을 만들려면 디스크를 준비해야 합니다. 디스크가 적절히 포맷되었는지 확인합니다.

- 슬라이스가 복사할 파일 시스템을 저장할 수 있을 정도의 충분한 크기를 갖고 있는지 확인합니다.
- 부트 환경 간에 복사가 아닌 공유하고자 하는 디렉토리를 포함하는 파일 시스템을 확인합니다. 디렉토리를 공유하고자 할 경우, 디렉토리를 자체 슬라이스에 놓고 새 부트 환경을 만들어야 합니다. 그러면 디렉토리는 다른 부트 환경과 공유될 수 있는 파일 시스템입니다. 공유용으로 파일 시스템을 만드는 자세한 내용은 **50 페이지 “공유 가능 파일 시스템에 대한 슬라이스 선택 지침”**을 참조하십시오.

RAID-1 볼륨(미러)를 만드는 경우의 Solaris Live Upgrade 요구 사항

Solaris Live Upgrade는 Solaris 볼륨 관리자 기술을 사용하여 부트 환경을 만들고, 여기에는 RAID-1 볼륨(미러)에 캡슐화된 파일 시스템이 포함됩니다. Solaris Live Upgrade가 Solaris 볼륨 관리자의 모든 기능을 구현하지는 않지만 Solaris 볼륨 관리자의 다음 구성 요소가 있어야 합니다.

표 3-3 Solaris Live Upgrade 및 RAID-1 볼륨에 필요한 구성 요소

요구 사항	설명	자세한 정보
최소한 하나의 상태 데이터베이스와 최소한 세 개의 상태 데이터베이스 복제본을 만들어야 합니다.	Solaris 볼륨 관리자 구성의 상태에 대한 디스크 정보를 저장하는 상태 데이터베이스입니다. 상태 데이터베이스는 복제된 여러 데이터베이스 복사본의 모음입니다. 각 복사본을 상태 데이터베이스 복제본이라고 합니다. 상태 데이터베이스가 복사되면 복제본은 단일 지점 오류로부터 데이터가 손실되지 않도록 보호합니다.	상태 데이터베이스 작성에 대한 자세한 내용은 Solaris Volume Manager Administration Guide 의 6 장, “ State Database (Overview) ”를 참조하십시오.
Solaris Live Upgrade는 오직 루트(/) 파일 시스템에 단일 슬라이스 연결이 포함된 RAID-1 볼륨(미러)만 지원합니다.	연결은 RAID-0 볼륨입니다. 슬라이스가 연결되면 첫 번째 사용 가능한 슬라이스가 가득 찰 때까지 해당 슬라이스에 데이터가 기록됩니다. 슬라이스가 가득 차면 데이터는 연속적으로 다음 슬라이스에 기록됩니다. 연결은 RAID-1 볼륨에 포함된 경우를 제외하고 데이터 중복을 제공하지 않습니다. RAID-1 볼륨은 최대 세 개의 연결로 구성될 수 있습니다.	미러된 파일 시스템을 만드는 지침은 48 페이지 “미러된 파일 시스템에 대한 슬라이스 선택 지침” 을 참조하십시오.

패키지 또는 패치로 시스템 업그레이드

Solaris Live Upgrade를 사용하여 시스템에 패치 및 패키지를 추가할 수 있습니다. Solaris Live Upgrade를 사용하는 경우 시스템을 다시 부트하는 경우에만 작동이 중지됩니다. `luupgrade` 명령을 사용하여 새 부트 환경에 패치와 패키지를 추가할 수 있습니다. `luupgrade` 명령을 사용하는 경우 Solaris Flash 아카이브를 사용하여 패치 또는 패키지를 설치할 수도 있습니다.



주의 - 패키지나 패치를 업그레이드, 추가 및 제거할 때, Solaris Live Upgrade에는 SVR4 고급 패키징 작성 지침을 준수하는 패키지 또는 패치가 필요합니다. Sun 패키지가 이러한 지침을 준수하는 반면 Sun은 타사 공급업체의 패키지 준수를 보장할 수 없습니다. 패키지가 이러한 지침을 위반하면 업그레이드하는 동안 패키지 추가 소프트웨어에 오류가 발생하거나 활성 부트 환경이 변경됩니다.

패키지 요구 사항에 대한 자세한 내용은 부록 B, “추가 SVR4 패키지 요구 사항(참조)”를 참조하십시오.

설치 유형	설명	자세한 정보
부트 환경에 패치 추가	새 부트 환경을 만들고 luupgrade 명령을 -t 옵션과 함께 사용합니다.	93 페이지 “부트 환경에서 네트워크 설치 이미지에 패치 추가”
부트 환경에 패키지 추가	luupgrade 명령을 -p 옵션과 함께 사용합니다.	92 페이지 “부트 환경에 네트워크 설치 이미지에 패키지 추가”
Solaris Live Upgrade를 사용하여 Solaris Flash 아카이브 설치	아카이브에는 이미 제공된 새 패키지와 패치가 포함된 부트 환경의 전체 복사본이 들어 있습니다. 이 사본은 다중 시스템에 설치할 수 있습니다.	<ul style="list-style-type: none"> ■ Solaris Flash 아카이브를 만드는 방법에 대한 자세한 내용은 Solaris 10 5/09 설치 설명서: Solaris Flash 아카이브(작성 및 설치)의 3 장, “Solaris Flash 아카이브 만들기(작업)”를 참조하십시오. ■ Solaris Live Upgrade를 사용하여 Solaris Flash 아카이브를 설치하는 방법은 102 페이지 “부트 환경에 Solaris Flash 아카이브 설치”를 참조하십시오.

업그레이드 및 패치 제한 사항

업그레이드 및 패치 제한 사항에 대한 자세한 내용은 [Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획](#)의 “업그레이드 및 패치 제한 사항”을 참조하십시오.

lucreate 명령을 사용하여 파일 시스템을 만드는 지침

lucreate -m 옵션은 새 부트 환경에 만들 파일 시스템과 파일 시스템 수를 지정합니다. 이 옵션을 반복하여 만들려는 파일 시스템의 정확한 수를 지정해야 합니다. -m 옵션을 사용하여 파일 시스템을 사용하는 경우 다음 지침을 따르십시오.

- 새 부트 환경에 대한 루트(/) 파일 시스템에 대해 한 개의 -m 옵션을 지정해야 합니다. -m 옵션 없이 lucreate를 실행할 경우 구성 메뉴가 표시됩니다. 구성 메뉴를 사용하면 파일을 새 마운트 지점으로 리디렉션하여 새 부트 환경을 사용자 지정할 수 있습니다.
- 현재 부트 환경에 존재하고 -m 옵션에서 지정되지 않은 모든 핵심 파일 시스템은 생성된 다음 최고 수준 파일 시스템으로 병합됩니다.

- `-m` 옵션에 의해 지정된 파일 시스템만이 새 부트 환경에서 생성됩니다. 현재 시스템에 있는 것과 같은 수의 파일 시스템을 만들려면 반드시 만들려는 각 파일 시스템에 대하여 하나의 `-m` 옵션을 지정해야 합니다.
예를 들어, `-m` 옵션을 한번 사용하면 모든 파일 시스템을 둘 위치를 지정합니다. 원본 부트 환경에서 모든 파일 시스템을 `-m` 옵션으로 지정한 단일 파일 시스템으로 병합합니다. `-m` 옵션을 두 번 지정하면 두 개의 파일 시스템이 생성됩니다. 루트(/), /opt 및 /var에 대한 파일 시스템이 있으면 새 부트 환경에서 각 파일 시스템에 대해 하나의 `-m` 옵션을 사용합니다.
- 마운트 지점을 중복하지 마십시오. 예를 들어, 두 개의 루트(/) 파일 시스템이 있을 수 없습니다.

파일 시스템에 대한 슬라이스 선택 지침

부트 환경에 대한 파일 시스템을 만드는 경우 규칙은 Solaris OS용 파일 시스템을 만들 때의 규칙과 동일합니다. Solaris Live Upgrade는 사용자가 중요한 파일 시스템에 대한 잘못된 구성을 만드는 것을 방지할 수 없습니다. 예를 들어 루트(/) 및 /kernel(루트(/) 파일 시스템의 잘못된 부분)을 위한 별도의 파일 시스템을 만들 수 있는 `lucreate` 명령을 입력할 수 있습니다.

디스크를 재분할하는 경우 슬라이스가 겹치면 안 됩니다. 이 조건에서는 새 부트 환경이 생성된 듯하지만 활성화되면 부트 환경이 부트되지 않습니다. 겹친 파일 시스템은 손상될 수 있습니다.

Solaris Live Upgrade가 제대로 작동하려면 활성 부트 환경의 `vfstab` 파일이 올바른 내용을 가져야 하고 최소한 루트(/) 파일 시스템에 대한 항목이 있어야 합니다.

루트(/) 파일 시스템에 대한 슬라이스 선택 지침

비활성 부트 환경을 만드는 경우 루트(/) 파일 시스템이 복사될 슬라이스를 식별해야 합니다. 루트(/) 파일 시스템에 대한 슬라이스를 선택할 때 다음 지침을 따르십시오. 슬라이스는 반드시 다음을 준수해야 합니다.

- 시스템이 부트할 수 있는 슬라이스여야 합니다.
- 권장 최소 크기를 충족해야 합니다.
- 다른 물리적 디스크 또는 루트(/) 파일 시스템과 동일한 디스크에 있을 수 있습니다.
- Veritas 볼륨 관리자 볼륨(VxVM)일 수 있습니다. 현재 시스템에 VxVM 볼륨이 구성되어 있으면 `lucreate` 명령을 사용하여 새 부트 환경을 만들 수 있습니다. 데이터를 새 부트 환경으로 복사하면 Veritas 파일 시스템 구성이 손실되며 새 부트 환경에서 UFS 파일 시스템이 작성됩니다.

미러된 파일 시스템에 대한 슬라이스 선택 지침

물리적 디스크 슬라이스, Solaris 볼륨 관리자 볼륨 또는 Veritas 볼륨 관리자 볼륨의 모든 조합이 포함된 새 부트 환경을 만들 수 있습니다. 새 부트 환경에 복사되는 중요한 파일 시스템은 다음 유형 중 하나입니다.

- 물리적 슬라이스
- RAID-1 볼륨(미러)에 포함된 단일 슬라이스 연결. 루트(/) 파일 시스템을 포함하는 슬라이스는 RAID-1 볼륨일 수 있습니다.
- RAID-0 볼륨에 포함된 단일 슬라이스 연결. 루트(/) 파일 시스템을 포함하는 슬라이스는 RAID-0 볼륨일 수 있습니다.

새 부트 환경을 작성할 때 `lucreate -m` 명령은 다음 세 가지 유형의 장치를 인식합니다.

- `/dev/dsk/cwt.xdysz` 형식의 물리적 슬라이스
- `/dev/md/dsk/dnum` 형식의 Solaris 볼륨 관리자 볼륨
- `/dev/vx/dsk/volume_name` 형식의 Veritas 볼륨 관리자 볼륨. 현재 시스템에 VxVM 볼륨이 구성되어 있으면 `lucreate` 명령을 사용하여 새 부트 환경을 만들 수 있습니다. 데이터를 새 부트 환경으로 복사하면 Veritas 파일 시스템 구성이 손실되며 새 부트 환경에서 UFS 파일 시스템이 작성됩니다.

주 - Veritas VxVM을 사용하여 업그레이드할 때 문제가 있는 경우 231 페이지 “Veritas VxVm을 실행하는 Solaris Live Upgrade를 사용하여 업그레이드하는 중에 시스템 장애 발생”을 참조하십시오.

RAID-1 볼륨(미러) 파일 시스템 생성 일반 지침

RAID-1 볼륨이 사용 중인지 또는 재동기화 중인지를 확인하거나 볼륨이 Solaris Live Upgrade 부트 환경에서 사용 중인 파일 시스템을 포함하는지를 확인하려면 다음 지침을 따르십시오.

볼륨 이름 지정 지침에 대한 자세한 내용은 **Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획의 “사용자 정의 JumpStart 및 Solaris Live Upgrade에 대한 RAID 볼륨 이름 요구 사항 및 지침”**을 참조하십시오.

볼륨 상태 확인

미러나 하위미러를 관리해야 하거나 사용 중인 경우에는 구성 요소를 분리할 수 없습니다. 새 부트 환경을 만들고 `detach` 키워드를 사용하기 전에 `metastat` 명령을 사용해야 합니다. `metastat` 명령은 미러를 재동기화하고 있는지 또는 미러를 사용 중인지를 확인합니다. 자세한 내용은 `metastat(1M)` 매뉴얼 페이지를 참조하십시오.

볼륨 분리 및 미러 재동기화

`detach` 키워드를 사용하여 하위미러를 분리할 경우, `lucreate`는 장치가 현재 다시 동기화 중인지 확인합니다. 장치가 재동기화 중인 경우 하위미러를 분리할 수 없으며 오류 메시지가 표시됩니다.

재동기화는 다음 문제가 발생한 경우 하위미러 간에 데이터를 복사하는 과정입니다.

- 하위미러 오류 발생
- 시스템 충돌
- 하위미러가 오프라인되었다가 다시 온라인 상태로 됨
- 새 하위미러 추가

재동기화에 대한 자세한 내용은 [Solaris Volume Manager Administration Guide](#)의 “RAID-1 Volume (Mirror) Resynchronization”을 참조하십시오.

Solaris 볼륨 관리자 명령 사용

비활성 부트 환경의 볼륨을 조작하려면 Solaris 볼륨 관리자 명령보다 `lucreate` 명령을 사용하십시오. Solaris 볼륨 관리자 소프트웨어는 부트 환경을 인식하지 못하지만 `lucreate` 명령에는 실수로 부트 환경을 삭제하지 못하도록 하는 확인 기능이 들어 있습니다. 예를 들어, `lucreate`는 Solaris 볼륨 관리자를 덮어쓰거나 삭제하지 못하게 합니다.

하지만 복잡한 Solaris 볼륨 관리자 연결, 스트라이프, 미러 등을 만드는 데 이미 Solaris 볼륨 관리자 소프트웨어를 사용했다면 Solaris 볼륨 관리자 소프트웨어를 사용하여 조작해야 합니다. Solaris Live Upgrade는 이러한 구성 요소를 인식하고 사용할 수 있도록 지원합니다. 볼륨 구성 요소를 만들거나 수정 또는 삭제할 수 있는 Solaris 볼륨 관리자 명령을 사용하려면 `lustatus` 또는 `lufslist` 명령을 사용합니다. 이러한 명령을 통해 Solaris Live Upgrade 부트 환경에서 사용되는 파일 시스템이 포함된 Solaris 볼륨 관리자 볼륨을 확인할 수 있습니다.

스왑 볼륨에 대한 슬라이스 선택 지침

이 지침에는 스왑 슬라이스용 구성 권장 사항과 예가 있습니다.

새 부트 환경에 대한 스왑 구성

`lucreate` 명령에 `-m` 옵션을 사용하여 다음 세 가지 방법으로 스왑 슬라이스를 구성할 수 있습니다.

- 스왑 슬라이스를 지정하지 않으면 현재 부트 환경에 속하는 스왑 슬라이스가 새 부트 환경에 구성됩니다.
- 하나 이상의 스왑 슬라이스를 지정하면 새 부트 환경에서는 해당 스왑 슬라이스만을 사용합니다. 두 부트 환경 간에는 스왑 슬라이스가 공유되지 않습니다.

- 스왑 슬라이스를 공유하도록 지정하고 스왑용 슬라이스를 새로 추가할 수 있습니다.

다음 예는 스왑을 구성하는 세 가지 방법을 보여 줍니다. 현재 부트 환경에서는 루트(/) 파일 시스템이 `c0t0d0s0`에 구성되어 있고 스왑 파일 시스템은 `c0t0d0s1`에 있습니다.

- 다음 예에서는 스왑 슬라이스가 지정되지 않았습니다. 새 부트 환경에서는 루트(/) 파일 시스템이 `c0t1d0s0`에 포함되어 있습니다. 스왑은 `c0t0d0s1`에 있으며 현재 부트 환경과 새 부트 환경 간에 공유됩니다.

```
# lucreate -n be2 -m /:/dev/dsk/c0t1d0s0:ufs
```

- 다음 예에서는 스왑 슬라이스가 지정되어 있습니다. 새 부트 환경에서는 루트(/) 파일 시스템이 `c0t1d0s0`에 포함되어 있습니다. 새 스왑 파일 시스템은 `c0t1d0s1`에 생성됩니다. 현재 부트 환경과 새 부트 환경 간에 스왑 슬라이스가 공유되지 않습니다.

```
# lucreate -n be2 -m /:/dev/dsk/c0t1d0s0:ufs -m -:/dev/dsk/c0t1d0s1:swap
```

- 다음 예에서는 스왑 슬라이스 하나가 추가되고 다른 스왑 슬라이스가 부트 환경 간에 공유됩니다. 새 부트 환경에서는 루트(/) 파일 시스템이 `c0t1d0s0`에 포함되어 있습니다. 새 스왑 슬라이스가 `c0t1d0s1`에 생성됩니다. `c0t0d0s1`의 스왑 슬라이스가 현재 부트 환경과 새 부트 환경 간에 공유됩니다.

```
# lucreate -n be2 -m /:/dev/dsk/c0t1d0s0:ufs -m -:shared:swap \
```

```
-m -:/dev/dsk/c0t1d0s1:swap
```

스왑이 사용 중인 경우 부트 환경 생성 실패

스왑 슬라이스가 현재 부트 환경을 제외한 부트 환경에서 사용되는 경우 부트 환경 만들기에 실패합니다. 부트 환경이 `-s` 옵션을 사용하여 만들어진 경우 대체 소스 부트 환경은 스왑 슬라이스를 사용할 수 있지만 다른 부트 환경은 사용할 수 없습니다.

공유 가능 파일 시스템에 대한 슬라이스 선택 지침

Solaris Live Upgrade는 슬라이스의 전체 내용을 지정된 새 부트 환경 슬라이스로 복사합니다. 해당 슬라이스의 일부 대형 파일 시스템을 복사하기보다 부트 환경 사이에 공유시켜 공간과 복사 시간을 절약할 수 있습니다. 루트(/) 및 `/var` 등의 OS에 중요한 파일 시스템은 반드시 복사되어야 합니다. `/home` 과 같은 파일 시스템은 중요한 파일 시스템이 아니며 부트 환경 사이에서 공유할 수 있습니다. 공유 가능 파일 시스템은 사용자 정의 파일 시스템이어야 하며 활성 및 새 부트 환경에서 별도의 스왑 슬라이스에 있어야 합니다. 필요에 따라 여러 방법으로 디스크를 재구성할 수 있습니다.

디스크 재구성	보기	자세한 정보
<p>새 부트 환경을 만들기 전에 디스크를 재분할하고 공유 가능 파일 시스템을 자체 슬라이에 둘 수 있습니다.</p>	<p>예를 들어 루트(/) 파일 시스템, /var 및 /home이 모두 동일한 슬라이에 있을 경우 디스크를 재구성하고 /home을 자체 슬라이에 배치합니다. 새 부트 환경을 작성할 때 /home은 기본적으로 새 부트 환경과 공유됩니다.</p>	<p>format(1M)</p>
<p>디렉토리를 공유하고자 하면 해당 디렉토리는 자체 슬라이드로 분리되어야 합니다. 그러면 해당 디렉토리는 다른 부트 환경과 공유할 수 있는 파일 시스템이 됩니다. <code>lucreate</code> 명령을 <code>-m</code> 옵션과 함께 사용하여 새 부트 환경을 만들고 디렉토리를 자체 슬라이드로 분리할 수 있습니다. 그러나 새 파일 시스템은 원래 부트 환경과 아직 공유할 수 없습니다. 다른 부트 환경을 만들려면 <code>lucreate</code> 명령을 <code>-m</code> 옵션과 함께 실행해야 합니다. 그러면 두 개의 새 부트 환경이 디렉토리를 공유할 수 있습니다.</p>	<p>예를 들어, Solaris 9 릴리스에서 Solaris 10 5/09 릴리스로 업그레이드하고 /home을 공유하려면 <code>-m</code> 옵션과 함께 <code>lucreate</code> 명령을 실행할 수 있습니다. 자체 슬라이드의 별도의 파일 시스템으로서 /home을 사용하여 Solaris 9 릴리스를 만들 수 있습니다. 그런 다음 <code>lucreate</code> 명령을 <code>-m</code> 옵션과 함께 다시 실행하여 해당 부트 환경을 복제합니다. 이렇게 하면 세 번째 부트 환경이 Solaris 10 5/09 릴리스로 업그레이드될 수 있습니다. /home은 Solaris 9와 Solaris 10 5/09 릴리스 간에 공유됩니다.</p>	<p>중요한 파일 시스템과 공유 가능한 파일 시스템에 대한 설명은 22 페이지 "파일 시스템 유형"을 참조하십시오.</p>

새 부트 환경 내용 사용자 정의

새 부트 환경을 만드는 경우 일부 디렉토리와 파일을 복사본에서 새 부트 환경으로 제외할 수 있습니다. 디렉토리를 제외한 경우 제외된 디렉토리의 하위 디렉토리 또는 파일을 지정하여 다시 포함시킬 수 있습니다. 복구된 하위 디렉토리나 파일은 새 부트 환경으로 복사됩니다. 예를 들어, /etc/mail에 있는 모든 파일과 디렉토리를 제외하고 /etc/mail/staff에 있는 모든 파일과 디렉토리를 포함하여 복사할 수 있습니다. 다음 명령은 staff 하위 디렉토리를 새 부트 환경에 복사합니다.

```
# lucreate -n second_disk -x /etc/mail -y /etc/mail/staff
```



주의 - 파일 제외 옵션은 주의해서 사용해야 합니다. 시스템에서 필요로 하는 파일이나 디렉토리를 제거하지 마십시오.

다음 표는 디렉토리와 파일을 제거하고 복원하는 `lucreate` 명령 옵션을 나열합니다.

지정 방법	제외 옵션	포함 옵션
디렉토리 또는 파일의 이름 지정	<code>-x exclude_dir</code>	<code>-y include_dir</code>
목록을 포함하는 파일 사용	<code>-f list_filename</code> <code>-z list_filename</code>	<code>-Y list_filename</code> <code>-z list_filename</code>

부트 환경을 만드는 경우 디렉토리 및 파일 사용자 정의 예는 82 페이지 “부트 환경 만들기 및 내용 사용자 정의”를 참조하십시오.

부트 환경 사이의 파일 동기화

새 부트 환경을 전환하여 활성화 상태로 만들 준비가 되면 새 부트 환경을 신속하게 활성화하고 재부트합니다. 새로 만든 부트 환경을 처음 부트할 때 부트 환경 간에 파일이 동기화됩니다. "동기화"란 특정 파일 시스템과 디렉토리가 마지막 활성화 부트 환경에서 부트 중인 부트 환경으로 복사됨을 의미합니다. 변경된 파일과 디렉토리를 복사합니다.

/etc/lu/synclist에 파일 추가

Solaris Live Upgrade는 중요한 파일이 변경되었는지 검사합니다. 이러한 파일의 내용이 양쪽 부트 환경 간에 같지 않으면 활성화 부트 환경에서 새 부트 환경으로 복사됩니다. 새 부트 환경을 만든 후에 변경했을 가능성이 있는 /etc/passwd 또는 /etc/group 파일과 같은 중요한 파일이 동기화의 대상입니다.

/etc/lu/synclist 파일에 동기화되는 디렉토리 및 파일 목록이 들어 있습니다. 경우에 따라 활성화 부트 환경의 다른 파일들을 새 부트 환경으로 복사하려고 할 수도 있습니다. 필요할 경우 디렉토리와 파일을 /etc/lu/synclist에 추가할 수 있습니다.

/etc/lu/synclist에 없는 파일을 추가하면 시스템이 부트되지 않을 수도 있습니다. 동기화 프로세스는 파일을 복사하고 디렉토리를 만드는 작업만 수행하고 파일과 디렉토리를 제거하지는 않습니다.

/etc/lu/synclist 파일의 다음 예는 이 시스템에 대해 동기화되는 표준 디렉토리 및 파일을 보여 줍니다.

/var/mail	OVERWRITE
/var/spool/mqueue	OVERWRITE
/var/spool/cron/crontabs	OVERWRITE
/var/dhcp	OVERWRITE
/etc/passwd	OVERWRITE
/etc/shadow	OVERWRITE
/etc/opasswd	OVERWRITE
/etc/oshadow	OVERWRITE
/etc/group	OVERWRITE
/etc/pwhist	OVERWRITE
/etc/default/passwd	OVERWRITE
/etc/dfs	OVERWRITE
/var/log/syslog	APPEND
/var/adm/messages	APPEND

synclist 파일에 추가하기에 적합한 디렉토리와 파일의 예는 다음과 같습니다.

/var/yp	OVERWRITE
/etc/mail	OVERWRITE
/etc/resolv.conf	OVERWRITE
/etc/domainname	OVERWRITE

synclist 파일 항목은 파일이나 디렉토리가 될 수 있습니다. 두 번째 필드는 부트 환경이 활성화될 때 수행되는 업데이트 방법을 나타냅니다. 세 가지 방법 중 선택하여 파일을 업데이트할 수 있습니다.

- **OVERWRITE** - 새 부트 환경 파일의 내용은 활성 부트 환경의 파일 내용으로 대체됩니다. 두 번째 필드에 동작이 지정되어 있지 않을 경우에는 **OVERWRITE**가 기본 동작입니다. 항목이 디렉토리이면 모든 하위 디렉토리가 복사됩니다. 모든 파일을 덮어씁니다. 새 부트 환경 파일의 날짜, 모드, 소유권은 이전 부트 환경의 해당 파일과 같습니다.
- **APPEND** - 새 부트 환경 파일 내용의 끝에 활성 부트 환경의 파일 내용이 추가됩니다. 이렇게 추가하면 파일에 있는 항목이 중복될 수 있습니다. 디렉토리에는 **APPEND**를 사용할 수 없습니다. 새 부트 환경 파일의 날짜, 모드, 소유권은 이전 부트 환경의 해당 파일과 같습니다.
- **PREPEND** - 새 부트 환경 파일의 시작 부분에 활성 부트 환경의 파일 내용이 추가됩니다. 이렇게 추가하면 파일에 있는 항목이 중복될 수 있습니다. 디렉토리에는 **PREPEND**를 사용할 수 없습니다. 새 부트 환경 파일의 날짜, 모드, 소유권은 이전 부트 환경의 해당 파일과 같습니다.

부트 환경 간의 강제 동기화

새로 만든 부트 환경으로 처음 부트하면 Solaris Live Upgrade가 새 부트 환경을 마지막 활성 부트 환경과 동기화합니다. 처음 부트하고 동기화한 다음에는 특별한 요청이 없는 한 동기화를 수행하지 않습니다. 강제로 동기화하려면 `-s` 옵션과 함께 `luactivate` 명령을 사용합니다.

관리하는 Solaris OS 버전이 여러 개인 경우 강제로 동기화를 수행할 수 있습니다. `email` 또는 `passwd/group` 같은 파일의 변경 사항을 활성화할 부트 환경에 적용할 수 있습니다. 강제로 동기화하면 Solaris Live Upgrade가 동기화 대상 파일 사이의 충돌을 검사합니다. 새 부트 환경이 부트되고 충돌이 감지되면 경고가 나타나고 파일이 동기화되지 않습니다. 이와 같은 충돌에도 불구하고 활성화를 성공적으로 완료할 수 있습니다. 새 부트 환경과 활성 부트 환경 모두에서 같은 파일을 변경하면 충돌이 일어날 수 있습니다. 예를 들어, 원본 부트 환경의 `/etc/passwd` 파일을 변경한 다음 새 부트 환경의 `/etc/passwd` 파일을 다른 내용으로 변경하면 동기화 프로세스는 동기화를 위해 복사할 파일을 선택할 수 없습니다.



주의 - 마지막 활성 부트 환경에서 발생했을 수 있는 변경 사항을 인식하거나 제어하지 못할 수 있으므로 이 옵션을 사용할 때에는 신중해야 합니다. 예를 들어, 현재 부트 환경에서 Solaris 10 5/09 소프트웨어를 실행하고 있으며 강제 동기화하여 Solaris 9 릴리스로 다시 부트하는 경우 파일이 Solaris 9 릴리스에서 변경될 수 있습니다. 파일은 OS의 릴리스에 따라 달라지므로 Solaris 9 릴리스로의 부트가 실패할 수 있는데, 이는 Solaris 10 5/09 파일이 Solaris 9 파일과 호환되지 않을 수 있기 때문입니다.

다중 부트 환경 부트

시스템에 둘 이상의 OS가 설치되어 있는 경우 SPARC 및 x86 플랫폼 모두에 대해 이러한 부트 환경에서 부트할 수 있습니다. 부트 시 사용할 수 있는 부트 환경에는 Solaris Live Upgrade 비활성 부트 환경이 포함됩니다.

- **Solaris 10 10/08 릴리스부터**(SPARC 기반 시스템용) ZFS 풀에서 ZFS 루트 파일 시스템을 부트할 수 있습니다. ZFS 루트 풀의 경우 boot 명령을 -L 옵션과 함께 사용하여 사용 가능한 부트 환경을 나열할 수 있습니다. 그런 다음 부트 환경을 선택하고 OBP boot 명령을 -Z 옵션과 함께 사용하여 해당 부트 환경을 부트할 수 있습니다. -z 옵션은 ZFS 루트 풀의 새 부트 환경을 부트하는 데 사용되기도 하는 luactivate 명령을 대체합니다. luactivate 명령은 부트 환경을 전환하는 데 사용되는 기본 방법입니다. UFS 파일 시스템의 경우 OpenBoot™ PROM OBP를 기본 관리 인터페이스로 계속 사용하며, OBP 명령을 통해 선택한 부트 옵션이 사용됩니다.
- **Solaris 10 1/06 릴리스부터**(x86 기반 시스템용) GRUB 부트 메뉴에서 서로 다른 부트 환경 간에 부트하는 데 필요한 인터페이스를 제공합니다. **Solaris 10 10/08 릴리스부터** 이 메뉴에는 부트 시 사용할 수 있는 ZFS 부트 환경이 나열됩니다. 기본 부트 환경이 ZFS 파일 시스템이고 GRUB 메뉴가 표시되는 경우에는 기본 부트 환경에서 부트하거나 다른 부트 환경에서 부트하도록 선택할 수 있습니다. GRUB 메뉴는 ZFS 루트 풀의 새 부트 환경을 부트하는 데 사용되기도 하는 luactivate 명령 사용을 대체합니다. luactivate는 부트 환경을 전환하는 데 사용되는 기본 방법입니다.

SPARC 및 x86 기반 시스템 모두에서 각 ZFS 루트 풀에는 기본 루트 파일 시스템으로 지정된 데이터 세트가 있습니다. SPARC의 경우 부트 명령을 입력하고, x86의 경우에는 GRUB 메뉴에서 기본값을 가져오면 이 기본 루트 파일 시스템이 부트됩니다.

주 - GRUB 메뉴가 Solaris Live Upgrade에서 설정한 메뉴를 제외하고 기본 메뉴 항목을 지정하도록 명시적으로 수정된 경우 해당 기본 메뉴 항목을 선택하면 풀의 기본 루트 파일 시스템의 부트가 발생하지 않을 수도 있습니다.

GRUB 부트 메뉴의 부트와 수정에 대한 자세한 내용은 다음 참조 항목을 참조하십시오.

작업	정보
GRUB 메뉴를 사용하여 부트 환경 활성화	112 페이지 “x86: GRUB 메뉴를 사용하여 부트 환경 활성화”
GRUB 메뉴를 사용하여 원래 부트 환경으로 풀백	118 페이지 “x86: GRUB 메뉴를 사용하여 성공적인 새 부트 환경 활성화 시 풀백”
SPARC 및 x86 정보와 부트 동작을 수정하고 부트하는 데 필요한 단계별 절차	<p>System Administration Guide: Basic Administration</p> <ul style="list-style-type: none"> ■ System Administration Guide: Basic Administration의 8 장, “Introduction to Shutting Down and Booting a System” ■ System Administration Guide: Basic Administration의 9 장, “Shutting Down and Booting a System (Overview)” ■ System Administration Guide: Basic Administration의 12 장, “Booting a Solaris System (Tasks)” ■ System Administration Guide: Basic Administration의 11 장, “Modifying Solaris Boot Behavior (Tasks)” ■ System Administration Guide: Basic Administration의 14 장, “Managing the Solaris Boot Archives (Tasks)”
ZFS 부트 환경 부트에 대한 개요와 단계별 절차	Solaris ZFS Administration Guide 의 “Booting From a ZFS Root File System”

Solaris Live Upgrade 문자 사용자 인터페이스

Sun에서는 더 이상 `lu` 명령의 사용을 권장하지 않습니다. `lu` 명령은 문자 사용자 인터페이스(CUI)를 표시합니다. CUI의 기본 명령 순서(일반적으로 `lucreate`, `luupgrade` 및 `luactivate` 명령)는 사용하기 편리합니다. 이 명령의 절차는 다음 장에 나와 있습니다.

Solaris Live Upgrade를 사용하여 부트 환경 만들기(작업)

이 장에서는 Solaris Live Upgrade 패키지 및 패치를 설치하고 부트 환경을 만드는 방법에 대해 설명합니다.

주 - 이 장에서는 UFS 파일 시스템용 Solaris Live Upgrade에 대해 설명합니다. UFS 파일 시스템을 ZFS 루트 풀로 마이그레이션하거나 ZFS 루트 풀을 만들고 설치하는 절차에 대한 자세한 내용은 13 장, “ZFS 루트 풀의 부트 환경 만들기”를 참조하십시오.

이 장은 다음 절로 구성되어 있습니다.

- 57 페이지 “작업 맵: Solaris Live Upgrade 설치 및 부트 환경 만들기”
- 58 페이지 “Solaris Live Upgrade 설치”
- 62 페이지 “새 부트 환경 생성”

작업 맵: Solaris Live Upgrade 설치 및 부트 환경 만들기

표 4-1 작업 맵: Solaris Live Upgrade 사용

작업	설명	수행 방법
Solaris Live Upgrade 패키지 설치	OS에 패키지 설치	58 페이지 “Solaris Live Upgrade 설치”
시스템에 패치 설치	Solaris Live Upgrade에 제한된 패치 개정판 세트가 필요합니다.	61 페이지 “Solaris Live Upgrade에 필요한 패치 설치”
부트 환경 만들기	비활성 부트 환경에 파일 시스템을 복사 및 재구성	62 페이지 “새 부트 환경 생성”

Solaris Live Upgrade 설치

Solaris Live Upgrade를 실행하기 전에 설치 매체에서 최신 Solaris Live Upgrade 패키지를 설치하고 SunSolveSM 정보 설명서 206844에 나열된 패치를 설치해야 합니다. 현재 OS에 Solaris Live Upgrade 패키지를 설치하고 이전 패키지를 제거해야 합니다. 반드시 Solaris Live Upgrade 패키지의 릴리스가 업그레이드하는 OS의 릴리스와 일치되어야 합니다. 예를 들어, 현재 OS가 Solaris 9 릴리스이며 Solaris 10 5/09 릴리스로 업그레이드하려는 경우 Solaris 10 5/09 릴리스의 Solaris Live Upgrade 패키지를 설치해야 합니다. 또한 SunSolve 정보 설명서 206844에 나열된 패치를 설치해야 합니다. 최신 패키지 및 패치를 설치하면 릴리스에 모든 최신 버그 수정과 새로운 기능이 포함됩니다. 새 부트 환경 만들기를 진행하기 전에 시스템과 관련된 모든 패치를 설치해야 합니다.

SunSolve 정보 설명서 206844에는 이전 패키지를 제거하고 새 패키지를 설치하는 방법이 설명되어 있으며 필요한 패치가 나열되어 있습니다. 다음 절차에서는 정보 설명서 206844에 설명된 절차에 대해 자세히 설명합니다.

- 61 페이지 “Solaris Live Upgrade에 필요한 패치 설치”
- 58 페이지 “pkgadd 명령을 사용하여 Solaris Live Upgrade 설치”
- 59 페이지 “Solaris 설치 프로그램을 사용하여 Solaris Live Upgrade 설치”

▼ pkgadd 명령을 사용하여 Solaris Live Upgrade 설치

설치 DVD 또는 CD에 있는 liveupgrade20 명령을 사용하거나 pkgadd 명령을 사용하여 패키지를 설치할 수 있습니다. liveupgrade20 명령을 사용하려면 Java 소프트웨어가 필요합니다. 시스템에 Java 소프트웨어를 설치하지 않은 경우 pkgadd 명령을 사용하여 패키지를 설치해야 합니다. 자세한 내용은 SunSolve 정보 설명서를 참조하십시오.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 기존 Solaris Live Upgrade 패키지를 제거합니다.

세 개의 Solaris Live Upgrade 패키지 SUNWluu, SUNWlur 및 SUNWlucfg는 Solaris Live Upgrade를 사용하여 업그레이드 또는 패치하기 위해 필요한 소프트웨어로 구성됩니다. 이들 패키지는 기존 소프트웨어, 새 기능, 버그 수정 등을 포함합니다. Solaris Live Upgrade를 사용하기 전에 시스템에서 기존 패키지를 제거하지 않고 새 패키지를 설치한 경우에는 대상 릴리스로 업그레이드 또는 패치할 수 없습니다. SUNWlucfg 패키지는 **Solaris 10 8/07 릴리스부터** 지원되는 새로운 기능입니다. Solaris 10 8/07 이전 릴리스의 Solaris Live Upgrade 패키지를 사용하는 경우에는 이 패키지를 제거하지 않아도 됩니다.

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

3 다음 순서에 따라 패키지를 설치합니다.

```
# pkgadd -d path_to_packages SUNWlucfg SUNWlur SUNWluu
```

`path_to_packages` 소프트웨어 패키지에 대한 절대 경로를 지정합니다.

- 4 패키지 자체가 제대로 설치되었는지 확인합니다.

```
# pkgchk -v SUNWLucfg SUNWlur SUNWluu
```

▼ Solaris 설치 프로그램을 사용하여 Solaris Live Upgrade 설치

설치 DVD 또는 CD에 있는 `liveupgrade20` 명령을 사용하여 패키지를 설치할 수 있습니다. `liveupgrade20` 명령을 사용하려면 Java 소프트웨어가 필요합니다. 시스템에 Java 소프트웨어를 설치하지 않은 경우 `pkgadd` 명령을 사용하여 패키지를 설치해야 합니다. 자세한 내용은 SunSolve 정보 설명서를 참조하십시오.

주 - 이 절차는 시스템이 **볼륨 관리자**를 실행 중이라고 가정합니다. 볼륨 관리자를 사용하여 이동식 매체를 관리하는 방법에 대한 자세한 내용은 [System Administration Guide: Devices and File Systems](#)를 참조하십시오.

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 기존 Solaris Live Upgrade 패키지를 제거합니다.

세 개의 Solaris Live Upgrade 패키지 `SUNWluu`, `SUNWlur` 및 `SUNWlucfg`는 Solaris Live Upgrade를 사용하여 업그레이드 또는 패치하기 위해 필요한 소프트웨어로 구성됩니다. 이들 패키지는 기존 소프트웨어, 새 기능, 버그 수정 등을 포함합니다. Solaris Live Upgrade를 사용하기 전에 시스템에서 기존 패키지를 제거하지 않고 새 패키지를 설치한 경우에는 대상 릴리스로 업그레이드 또는 패치할 수 없습니다. `SUNWlucfg` 패키지는 **Solaris 10 8/07 릴리스**부터 지원되는 새로운 기능입니다. Solaris 10 8/07 이전 릴리스의 Solaris Live Upgrade 패키지를 사용하는 경우에는 이 패키지를 제거하지 않아도 됩니다.

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- 3 Solaris Operating System DVD 또는 Solaris Software - 2 CD를 넣습니다.

- 4 사용 중인 매체에 대한 설치 프로그램을 실행합니다.

- Solaris Operating System DVD를 사용 중인 경우, 설치 프로그램으로 디렉토리를 변경하고 설치 프로그램을 실행합니다.

```
# cd /cdrom/cdrom0/Solaris_10/Tools/Installers
# ./liveupgrade20
```

Solaris 설치 프로그램 GUI가 표시됩니다. 스크립트를 사용할 경우 `-noconsole` 및 `-nodisplay` 옵션을 사용하여 GUI가 표시되지 않게 할 수 있습니다.

- Solaris Software - 2 CD를 사용 중인 경우, 설치 프로그램을 실행합니다.

```
% ./installer
```

Solaris 설치 프로그램 GUI가 표시됩니다.

- 5 설치 유형 선택 패널에서 사용자 정의를 누릅니다.
- 6 로컬 선택 패널에서 설치할 언어를 누릅니다.
- 7 설치할 소프트웨어를 선택합니다.
 - DVD의 경우, 구성 요소 선택 패널에서 다음을 눌러 패키지를 설치합니다.
 - CD의 경우, 제품 선택 패널에서 Solaris Live Upgrade 기본 설치를 누르고 다른 소프트웨어를 눌러 해당 선택을 해제합니다.
- 8 Solaris 설치 프로그램 패널에 표시되는 설명을 따라 소프트웨어를 설치합니다. 필요한 패치를 설치할 준비가 되었습니다.

Solaris Live Upgrade에 필요한 패치 설치

설명	자세한 정보
<p>주의 - Solaris Live Upgrade가 올바르게 작동하려면 특정 OS 버전에 대한 제한된 패치 개정판 세트를 설치해야 합니다. Solaris Live Upgrade를 설치하거나 실행하기 전에 이러한 패치를 설치해야 합니다.</p>	<p>http://sunsolve.sun.com을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 206844(이전의 72099)를 검색합니다.</p>
<p>x86 전용 - 이 패치 세트가 설치되어 있지 않으면 Solaris Live Upgrade가 실패하며 다음 오류 메시지가 표시될 수 있습니다. 다음 오류 메시지가 표시되지 않아도 필요한 패치가 아직 설치되어 있지 않을 수 있습니다. Solaris Live Upgrade를 설치하기 전에 항상 SunSolve info doc에 나열된 모든 패치가 설치되어 있는지 확인하십시오.</p>	
<pre>ERROR: Cannot find or is not executable: </sbin/biosdev>. ERROR: One or more patches required by Live Upgrade has not been installed.</pre>	
<p>정보 문서 206844(이전의 72099)에 나열된 패치는 언제든지 변경될 수 있습니다. 이러한 패치는 Solaris Live Upgrade의 결함과 Solaris Live Upgrade가 사용하는 구성 요소의 결함을 해결할 수 있습니다. Solaris Live Upgrade에서 문제가 발생하면 최신 Solaris Live Upgrade 패치가 설치되어 있는지 확인합니다.</p>	
<p>Solaris 8 또는 Solaris 9 OS를 실행하는 경우 Solaris Live Upgrade 설치 프로그램을 실행하지 못할 수 있습니다. 이 릴리스에는 Java 2 런타임 환경을 실행하는 데 필요한 패치 집합이 포함되어 있지 않습니다. Solaris Live Upgrade 설치 프로그램을 실행하고 패키지를 설치하려면 반드시 Java 2 런타임 환경으로 권장되는 패치가 있어야 합니다.</p>	<p>Solaris Live Upgrade 패키지를 설치하려면 pkgadd 명령을 사용합니다. 또는 권장되는 패치 클러스터를 Java 2 런타임 환경용으로 설치합니다. 패치 클러스터는 http://sunsolve.sun.com에 있습니다.</p>

▼ 필수 패치 설치

- 1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 패치를 로컬 디스크에 저장할 경우 /var/tmp/lupatches와 같은 디렉토리를 만듭니다.
- 3 SunSolve 웹 사이트에서 패치 목록을 확인합니다.

- 4 이 예에서처럼 패치 디렉토리를 변경합니다.

```
# cd /var/tmp/lupatches
```

- 5 patchadd 명령을 사용하여 패치를 설치합니다.

```
# patchadd path_to_patches patch-id patch-id
```

patch-id는 패치 번호입니다. 공백을 사용하여 여러 패치 이름을 구분합니다.

주 - 정보 설명서 206844에 지정된 순서대로 패치를 적용해야 합니다.

- 6 필요한 경우 시스템을 재부트합니다. 특정 패치는 재부트해야만 적용됩니다.

x86에만 해당: 시스템을 재부트해야 하며, 그렇지 않으면 Solaris Live Upgrade가 실패합니다.

```
# init 6
```

이제 새 부트 환경을 성공적으로 만드는 데 필요한 패키지 및 패치가 설치되었습니다.

새 부트 환경 생성

부트 환경 만들기는 중요 파일 시스템을 활성 부트 환경에서 새 부트 환경으로 복사하는 방법을 제공합니다. lucreate 명령을 사용하면 필요한 경우 디스크를 다시 구성하고 파일 시스템을 사용자 정의하며, 중요 파일 시스템을 새 부트 환경으로 복사할 수 있습니다.

파일 시스템을 새 부트 환경으로 복사하기 전에 중요한 파일 시스템 디렉토리가 부모 디렉토리로 병합되거나 부모 디렉토리에서 분리되도록 사용자 정의할 수 있습니다. 사용자 정의(공유 가능) 파일 시스템은 기본적으로 부트 환경 사이에서 공유됩니다. 그러나 공유 가능 파일 시스템은 필요할 경우 복사할 수 있습니다. 스왑(공유 가능 볼륨)도 분리 및 병합할 수 있습니다. 중요한 파일 시스템과 공유 가능한 파일 시스템에 대한 개요는 22 페이지 “파일 시스템 유형”을 참조하십시오.

주 - 이 장에서는 UFS 파일 시스템용 Solaris Live Upgrade에 대해 설명합니다. UFS 파일 시스템을 ZFS 루트 풀로 마이그레이션하거나 ZFS 루트 풀을 만들고 설치하는 절차에 대한 자세한 내용은 13 장, “ZFS 루트 풀의 부트 환경 만들기”를 참조하십시오.

▼ 처음으로 부트 환경 만들기

-m 옵션과 같이 사용되는 lucreate 명령은 새 부트 환경에서 작성될 파일 시스템과 파일 시스템의 수를 지정합니다. 이 옵션을 반복하여 만들려는 파일 시스템의 정확한 수를 지정해야 합니다. 예를 들어, -m 옵션을 한번 사용하면 모든 파일 시스템을 둘 위치를 지정합니다. 원본 부트 환경에서 모든 파일 시스템을 -m 옵션으로 지정한 단일 파일

시스템으로 병합합니다. `-m` 옵션을 두 번 지정하면 두 개의 파일 시스템이 생성됩니다. `-m` 옵션을 사용하여 파일 시스템을 사용하는 경우 다음 지침을 따르십시오.

- 새 부트 환경에 대한 루트(/) 파일 시스템에 대해 한 개의 `-m` 옵션을 지정해야 합니다. `-m` 옵션 없이 `lucreate`를 실행할 경우 구성 메뉴가 표시됩니다. 구성 메뉴를 사용하면 파일을 새 마운트 지점으로 리디렉션하여 새 부트 환경을 사용자 지정할 수 있습니다.
- 현재 부트 환경에 존재하고 `-m` 옵션에서 지정되지 않은 모든 중요한 파일 시스템은 그 다음 최고 수준의 파일 시스템으로 병합됩니다.
- `-m` 옵션에 의해 지정된 파일 시스템만이 새 부트 환경에서 생성됩니다. 현재 부트 환경이 다중 파일 시스템을 포함하고 생성된 새 부트 환경에 동일한 수의 파일 시스템이 있게 하려면 만들 각 파일 시스템에 대하여 하나의 `-m` 옵션을 지정해야 합니다. 예를 들어, 루트(/), `/opt` 및 `/var`에 대한 파일 시스템이 있으면 새 부트 환경에서 각 파일 시스템에 대해 한 개의 `-m` 옵션을 사용합니다.
- 마운트 지점을 중복하지 마십시오. 예를 들어, 두 개의 루트(/) 파일 시스템이 있을 수 없습니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 부트 환경을 만들려면 다음을 입력합니다.

```
# lucreate [-A 'BE_description'] -c BE_name \
-m mountpoint:device[,metadevice]:fs_options [-m ...] -n BE_name
```

`-A 'BE_description'`

(선택 사항) 부트 환경 이름(`BE_name`)과 연관된 부트 환경 설명을 작성할 수 있도록 합니다. 설명은 길이 제한이 없고 모든 문자를 포함할 수 있습니다.

`-c BE_name`

활성 부트 환경에 이름 `BE_name`을 할당합니다. 이 옵션은 필수 사항이 아니며 부트 환경을 처음 만들 때만 사용됩니다. 처음 `lucreate`를 실행할 때 `-c` 옵션을 생략하면 소프트웨어에서는 기본 파일 이름을 사용합니다.

다음 기준에 따라 기본 이름을 선택할 수 있습니다.

- 물리적 부트 장치를 확인할 수 있는 경우에는 해당 장치의 기본 이름을 사용하여 현재 부트 환경의 이름을 지정합니다.

예를 들어, 물리적 부트 장치가 `/dev/dsk/c0t0d0s0`이면 현재 부트 환경의 이름이 `c0t0d0s0`으로 지정됩니다.

- 물리적 부트 장치를 결정할 수 없는 경우 `uname` 명령의 이름을 `-s` 및 `-r` 옵션을 조합하여 만듭니다.

예를 들어 `uname -s`가 `SunOS`라는 OS 이름을 반환하고 `uname -r`가 `5.9`라는 릴리스 이름을 반환할 경우 현재 부트 환경에 `SunOS5.9`라는 이름이 지정됩니다.

- 위 두 가지를 모두 사용해도 이름을 확인할 수 없으면 `current`를 사용하여 현재 부트 환경 이름을 지정합니다.

주 - 처음 부트 환경을 만든 다음 `-c` 옵션을 사용하는 경우 해당 옵션은 무시되거나 오류 메시지가 표시됩니다.

- 지정한 이름이 현재 부트 환경 이름과 같으면 이 옵션이 무시됩니다.
- 지정한 이름이 현재 부트 환경 이름과 다르면 오류 메시지가 표시되고 만들기가 실패합니다. 다음 예는 오류 메시지가 발생하는 부트 환경 이름을 보여 줍니다.

```
# lucurr
c0t0d0s0
# lucreate -c /dev/dsk/clt1d1s1 -n newbe -m /:/dev/dsk/clt1d1s1:ufs
ERROR: current boot environment name is c0t0d0s0: cannot change
name using <-c clt1d1s1>
```

`-m mountpoint:device[,metadevice]:fs_options [-m...]`

`fstab`에 새 부트 환경의 파일 시스템 구성을 지정합니다. `-m`에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- `mountpoint`는 모든 유효한 마운트 지점 또는 `-`(하이픈)일 수 있으며 스왑 분할 영역을 나타냅니다.
- `device` 필드는 다음 중 하나일 수 있습니다.
 - `/dev/dsk/cwtxdysz` 형식의 디스크 장치 이름
 - `/dev/md/dsk/dnum` 형식의 Solaris 볼륨 관리자 볼륨 이름
 - `/dev/md/vxfs/dsk/dnum` 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 `merged`는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냄.
- `fs_options` 필드는 다음 중 하나일 수 있습니다.
 - `ufs`, UFS 파일 시스템을 나타냄.
 - `vxfs`, Veritas 파일 시스템을 나타냄.
 - `swap`, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 `-`(하이픈)이어야 합니다.
 - 로컬 장치(미러)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다. 이들 키워드에 대한 설명은 77 페이지 “RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기”을 참조하십시오.

`-n BE_name`

생성될 부트 환경의 이름 `BE_name`은 시스템에서 고유해야 합니다.

새 부트 환경 만들기가 완료되면 업그레이드 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

예 4-1 부트 환경 만들기

이 예에서 활성 부트 환경의 이름은 `first_disk`로 지정됩니다. 파일 시스템에 대한 마운트 지점은 `-m` 옵션을 사용하여 나타냅니다. 두 개의 파일 시스템, 루트(/)와 /usr이 생성됩니다. 새 부트 환경 이름은 `second_disk`로 지정됩니다. 설명(`mydescription`)은 `second_disk`와 연결됩니다. 새 부트 환경인 `second_disk`에서 스왑은 소스인 `first_disk`에서 자동으로 공유됩니다.

```
# lucreate -A 'mydescription' -c first_disk -m /:/dev/dsk/c0t4d0s0:ufs \
-m /usr:/dev/dsk/c0t4d0s3:ufs -n second_disk
```

▼ 부트 환경 만들기 및 파일 시스템 병합

주 `-m` 옵션과 함께 `-lucreate` 명령을 사용하여 새 부트 환경에 만들 파일 시스템과 파일 시스템 수를 지정할 수 있습니다. 이 옵션을 반복하여 만들려는 파일 시스템의 정확한 수를 지정해야 합니다. 예를 들어, `-m` 옵션을 한번 사용하면 모든 파일 시스템을 둘 위치를 지정합니다. 원래 부트 환경의 모든 파일 시스템을 하나의 파일 시스템으로 병합합니다. `-m` 옵션을 두 번 지정하면 두 개의 파일 시스템이 생성됩니다.

1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 다음을 입력합니다.

```
# lucreate -A 'BE_description' \
-m mountpoint:device[,metadevice]:fs_options \
-m [...] -m mountpoint:merged:fs_options -n BE_name
```

`-A BE_description`

(선택 사항) 부트 환경 이름(`BE_name`)과 연관된 부트 환경 설명을 작성할 수 있도록 합니다. 설명은 길이 제한이 없고 모든 문자를 포함할 수 있습니다.

`-m mountpoint:device[,metadevice]:fs_options [-m...]`

새 부트 환경의 파일 시스템의 구성을 지정합니다. `-m`에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- `mountpoint`는 모든 유효한 마운트 지점 또는 `-`(하이픈)일 수 있으며 스왑 분할 영역을 나타냅니다.

- *device* 필드는 다음 중 하나일 수 있습니다.
 - /dev/dsk/cwtxdysz 형식의 디스크 장치 이름
 - /dev/md/dsk/dnum 형식의 Solaris 볼륨 관리자 메타 장치 이름
 - /dev/vx/dsk/volume_name 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 merged는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냄.
- *fs_options* 필드는 다음 중 하나일 수 있습니다.
 - ufs, UFS 파일 시스템을 나타냄.
 - vxfs, Veritas 파일 시스템을 나타냄.
 - swap, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 -(하이픈)이어야 합니다.
 - 로컬 장치(미러)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다. 이들 키워드에 대한 설명은 77 페이지 “RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기”을 참조하십시오.

-n *BE_name*

생성될 부트 환경의 이름 *BE_name*은 시스템에서 고유해야 합니다.

새 부트 환경 만들기가 완료되면 업그레이드 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

예 4-2 부트 환경 만들기 및 파일 시스템 병합

이 예에서 현재 부트 환경의 파일 시스템은 루트(/), /usr 및 /opt입니다. /opt 파일 시스템은 부모 파일 시스템 /usr과 결합됩니다. 새 부트 환경 이름은 second_disk로 지정됩니다. 설명(mydescription)은 이름 second_disk와 연결됩니다.

```
# lucreate -A 'mydescription' -c first_disk \
-m /:/dev/dsk/c0t4d0s0:ufs -m /usr:/dev/dsk/c0t4d0s1:ufs \
-m /usr/opt:merged:ufs -n second_disk
```

▼ 부트 환경 만들기 및 파일 시스템 분할

주 - 부트 환경에 대한 파일 시스템을 만드는 경우 규칙은 Solaris OS용 파일 시스템을 만들 때의 규칙과 동일합니다. Solaris Live Upgrade는 중요한 파일 시스템에 잘못된 구성을 만드는 것을 방지할 수 없습니다. 예를 들어 루트(/) 및 /kernel(루트(/) 파일 시스템의 잘못된 부분)을 위한 별도의 파일 시스템을 만들 수 있는 lucreate 명령을 입력할 수 있습니다.

여러 마운트 지점으로 디렉토리를 분리할 때 하드 링크는 파일 시스템 간에 유지되지 않습니다. 예를 들어, /usr/stuff1/file이 /usr/stuff2/file에 하드 링크되고 /usr/stuff1과 /usr/stuff2가 별도의 파일 시스템으로 분리되면 파일 간의 링크는 더 이상 존재하지 않습니다. lucreate는 경고 메시지를 표시하고 심볼릭 링크가 생성되어 손실된 하드 링크를 교체합니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 다음을 입력합니다.

```
# lucreate [-A 'BE_description'] \
  -m mountpoint:device[,metadevice]:fs_options \
  -m mountpoint:device[,metadevice]:fs_options -n new_BE
-A 'BE_description'
```

(선택 사항) 부트 환경 이름(BE_name)과 연관된 부트 환경 설명을 작성할 수 있도록 합니다. 설명은 길이 제한이 없고 모든 문자를 포함할 수 있습니다.

```
-m mountpoint:device[,metadevice]:fs_options [-m...]
```

새 부트 환경의 파일 시스템의 구성을 지정합니다. -m에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- *mountpoint*는 모든 유효한 마운트 지점 또는 -(하이픈)일 수 있으며 스왑 분할 영역을 나타냅니다.
- *device* 필드는 다음 중 하나일 수 있습니다.
 - /dev/dsk/cwtxdysz 형식의 디스크 장치 이름
 - /dev/md/dsk/dnum 형식의 Solaris 볼륨 관리자 메타 장치 이름
 - /dev/vx/dsk/volume_name 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 merged는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냄.
- *fs_options* 필드는 다음 중 하나일 수 있습니다.
 - ufs, UFS 파일 시스템을 나타냄.
 - vxfs, Veritas 파일 시스템을 나타냄.
 - swap, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 -(하이픈)이어야 합니다.
 - 로컬 장치(미러)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다. 이들 키워드에 대한 설명은 [77 페이지 “RAID-1 볼륨\(미러\)을 사용하여 부트 환경 만들기](#)”를 참조하십시오.

-n *BE_name*

생성될 부트 환경의 이름 *BE_name*은 시스템에서 고유해야 합니다.

예 4-3 부트 환경 만들기 및 파일 시스템 분할

이 예에서 선행 명령이 새 부트 환경에서 여러 디스크 슬라이스에 걸쳐 루트(/) 파일 시스템을 분리합니다. 소스 부트 환경에서 /usr, /var 및 /opt가 루트(/) /dev/dsk/c0t0d0s0 에 있다고 가정합니다.

새 부트 환경에서 다음과 같이 /usr, /var 및 /opt를 분리하여 이들 파일 시스템을 자체 슬라이스에 마운트합니다.

```
/dev/dsk/c0t1d0s0 /
/dev/dsk/c0t1d0s1 /var
/dev/dsk/c0t1d0s7 /usr
/dev/dsk/c0t1d0s5 /opt
```

설명(mydescription)은 부트 환경 이름인 second_disk와 연결됩니다.

```
# lucreate -A 'mydescription' -c first_disk \
-m /:/dev/dsk/c0t1d0s0:ufs -m /usr:/dev/dsk/c0t1d0s7:ufs \
-m /var:/dev/dsk/c0t1d0s1:ufs -m /opt:/dev/dsk/c0t1d0s5:ufs \
-n second_disk
```

새 부트 환경 만들기가 완료되면 업그레이드 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

▼ 부트 환경 만들기 및 스왑 재구성

기본적으로 교체 슬라이스는 부트 환경 사이에서 공유됩니다. 스왑을 -m 옵션과 함께 지정하지 않으면 현재 부트 환경과 새 부트 환경이 동일한 스왑 슬라이스를 공유합니다. 새 부트 환경의 스왑을 재구성하려면 -m 옵션을 사용하여 새 부트 환경에서 스왑 슬라이스를 추가 또는 제거합니다.

주 - 스왑 슬라이스는 현재 부트 환경을 제외한 모든 부트 환경 또는 -s 옵션이 사용될 경우 소스 부트 환경에서 사용될 수 없습니다. 스왑 슬라이스가 다른 부트 환경, 스왑, UFS 또는 다른 파일 시스템에 의해 사용되고 있을 경우 부트 환경 만들기가 실패합니다.

기존 스왑 슬라이스로 부트 환경을 만들고 `vfstab` 파일을 만든 후에 편집할 수 있습니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 다음을 입력합니다.

```
# lucreate [-A 'BE_description'] \
  -m mountpoint:device[,metadevice]:fs_options \
  -m -:device:swap -n BE_name
```

-A 'BE_description'

(선택 사항) 부트 환경 이름(BE_name)과 연관된 부트 환경 설명을 작성할 수 있도록 합니다. 설명은 길이 제한이 없고 모든 문자를 포함할 수 있습니다.

-m mountpoint:device[,metadevice]:fs_options [-m...]

새 부트 환경의 파일 시스템의 구성을 지정합니다. -m에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- *mountpoint*는 모든 유효한 마운트 지점 또는 -(하이픈)일 수 있으며 스왑 분할 영역을 나타냅니다.
- *device* 필드는 다음 중 하나일 수 있습니다.
 - /dev/dsk/cwtxdysz 형식의 디스크 장치 이름
 - /dev/md/dsk/dnum 형식의 Solaris 볼륨 관리자 메타 장치 이름
 - /dev/vx/dsk/volume_name 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 merged는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냄.
- *fs_options* 필드는 다음 중 하나일 수 있습니다.
 - ufs, UFS 파일 시스템을 나타냄.
 - vxfs, Veritas 파일 시스템을 나타냄.
 - swap, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 -(하이픈)이어야 합니다.
 - 로컬 장치(미리)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다. 이들 키워드에 대한 설명은 77 페이지 “RAID-1 볼륨(미리)을 사용하여 부트 환경 만들기”을 참조하십시오.

-n BE_name

생성될 부트 환경의 이름 BE_name은 고유해야 합니다.

새 부트 환경은 다른 슬라이스 또는 장치로 이동된 스왑으로 만듭니다.

새 부트 환경 만들기가 완료되면 업그레이드 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

예 4-4 부트 환경 만들기 및 스왑 재구성

이 예에서 현재 부트 환경은 `/dev/dsk/c0t0d0s0`의 루트(/)를 포함하고 스왑은 `/dev/dsk/c0t0d0s1`에 있습니다. 새 부트 환경은 루트(/)를 `/dev/dsk/c0t4d0s0`으로 복사하고 `/dev/dsk/c0t0d0s1`과 `/dev/dsk/c0t4d0s1`을 모두 스왑 슬라이스로 사용합니다. 설명(mydescription)은 부트 환경 이름인 `second_disk`와 연결됩니다.

```
# lucreate -A 'mydescription' -c first_disk \
-m /:/dev/dsk/c0t4d0s0:ufs -m -:/dev/dsk/c0t0d0s1:swap \
-m -:/dev/dsk/c0t4d0s1:swap -n second_disk
```

이러한 스왑 할당은 `second_disk`에서 부트 후에만 유효합니다. 긴 목록의 스왑 슬라이스가 있으면 `-M` 옵션을 사용합니다. 70 페이지 “목록을 사용하여 부트 환경 만들기 및 스왑 재구성”을 참조하십시오.

▼ 목록을 사용하여 부트 환경 만들기 및 스왑 재구성

긴 목록의 교체 슬라이스가 있으면 교체 목록을 작성합니다. `lucreate`는 새 부트 환경의 스왑 슬라이스에 대해 이 목록을 사용합니다.

주-스왑 슬라이스는 현재 부트 환경을 제외한 모든 부트 환경 또는 `-s` 옵션이 사용될 경우 소스 부트 환경에서 사용될 수 없습니다. 스왑 슬라이스가 다른 부트 환경, 스왑, UFS 또는 다른 파일 시스템을 포함하는지 여부에 관계없이 다른 부트 환경에 의해 사용되고 있을 경우 부트 환경 만들기가 실패합니다.

- 1 새 부트 환경에서 사용할 스왑 슬라이스의 목록을 만듭니다. 이 파일의 위치와 이름은 사용자가 정의합니다. 이 예에서 `/etc/lu/swapslices` 파일에는 장치 및 슬라이스 목록이 들어 있습니다.

```
-:/dev/dsk/c0t3d0s2:swap
-:/dev/dsk/c0t3d0s2:swap
-:/dev/dsk/c0t4d0s2:swap
-:/dev/dsk/c0t5d0s2:swap
-:/dev/dsk/c1t3d0s2:swap
-:/dev/dsk/c1t4d0s2:swap
-:/dev/dsk/c1t5d0s2:swap
```

- 2 다음을 입력합니다.

```
# lucreate [-A 'BE_description'] \
-m mountpoint:device[,metadevice]:fs_options \
-M slice_list -n BE_name
```

```
-A 'BE_description'
```

(선택 사항) 부트 환경 이름(BE_name)과 연관된 부트 환경 설명을 작성할 수 있도록 합니다. 설명은 길이 제한이 없고 모든 문자를 포함할 수 있습니다.

`-m mountpoint:device[,metadevice]:fs_options [-m...]`

새 부트 환경의 파일 시스템의 구성을 지정합니다. `-m`에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- `mountpoint`는 모든 유효한 마운트 지점 또는 `-(하이픈)`일 수 있으며 스왑 분할 영역을 나타냅니다.
- `device` 필드는 다음 중 하나일 수 있습니다.
 - `/dev/dsk/cwtxdysz` 형식의 디스크 장치 이름
 - `/dev/md/dsk/dnum` 형식의 Solaris 볼륨 관리자 메타 장치 이름
 - `/dev/vx/dsk/volume_name` 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 `merged`는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냄.
- `fs_options` 필드는 다음 중 하나일 수 있습니다.
 - `ufs`, UFS 파일 시스템을 나타냄.
 - `vxfs`, Veritas 파일 시스템을 나타냄.
 - `swap`, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 `-(하이픈)`이어야 합니다.
 - 로컬 장치(미러)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다. 이들 키워드에 대한 설명은 77 페이지 “RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기”를 참조하십시오.

`-M slice_list`

파일 `-slice_list`에서 수집되는 `m` 옵션의 목록. `-m`에 대하여 지정되는 형식으로 이 인자를 지정합니다. 해시 표시(`#`)로 시작하는 명령줄은 무시됩니다. `-M` 옵션은 부트 환경에 대한 파일 시스템의 긴 목록이 있을 경우 유용합니다. `-m`과 `-M` 옵션을 결합할 수 있습니다. 예를 들어, `slice_list`에 스왑 슬라이스를 저장하고 루트(`/`) 및 `/usr` 슬라이스를 `-m`으로 지정할 수 있습니다.

`-m` 및 `-M` 옵션은 특정 마운트 지점에 대한 복수 슬라이스의 목록을 지원합니다. 이러한 슬라이스를 처리할 때 `lucreate`는 사용할 수 없는 슬라이스는 건너뛰고 사용 가능한 첫 번째 슬라이스를 선택합니다.

`-n BE_name`

생성될 부트 환경의 이름 `BE_name`은 고유해야 합니다.

새 부트 환경 만들기가 완료되면 업그레이드 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

예 4-5 목록을 사용하여 부트 환경 만들기 및 스왑 재구성

이 예에서 새 부트 환경의 스왑은 `/etc/lu/swapslices` 파일에 기록된 슬라이스 목록입니다. 설명(`mydescription`)은 `second_disk`와 연결됩니다.

```
# lucreate -A 'mydescription' -c first_disk \
-m /:/dev/dsk/c02t4d0s0:ufs -m /usr:/dev/dsk/c02t4d0s1:ufs \
-M /etc/lu/swapslices -n second_disk
```

▼ 부트 환경 만들기 및 공유 가능 파일 시스템 복사

공유 가능 파일 시스템이 새 부트 환경으로 복사되게 하려면 `-m` 옵션으로 복사될 마운트 지점을 지정합니다. 그렇지 않은 경우 공유 가능한 파일 시스템이 기본적으로 공유되며 `vfstab` 파일의 동일한 마운트 지점이 유지됩니다. 공유 가능 파일 시스템에 적용된 모든 업데이트는 양쪽 부트 환경 모두에서 사용 가능합니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 부트 환경을 만듭니다.

```
# lucreate [-A 'BE_description'] \
-m mountpoint:device[,metadevice]:fs_options \
-m mountpoint:device[,metadevice]:fs_options -n BE_name
```

-A 'BE_description'

(선택 사항) 부트 환경 이름(BE_name)과 연관된 부트 환경 설명을 작성할 수 있도록 합니다. 설명은 길이 제한이 없고 모든 문자를 포함할 수 있습니다.

-m mountpoint:device[,metadevice]:fs_options [-m...]

새 부트 환경의 파일 시스템의 구성을 지정합니다. `-m`에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- `mountpoint`는 모든 유효한 마운트 지점 또는 -(하이픈)일 수 있으며 스왑 분할 영역을 나타냅니다.
- `device` 필드는 다음 중 하나일 수 있습니다.
 - `/dev/dsk/cwtxdysz` 형식의 디스크 장치 이름
 - `/dev/md/dsk/dnum` 형식의 Solaris 볼륨 관리자 메타 장치 이름
 - `/dev/vx/dsk/volume_name` 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 `merged`는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냄.
- `fs_options` 필드는 다음 중 하나일 수 있습니다.
 - `ufs`, UFS 파일 시스템을 나타냄.
 - `vxfs`, Veritas 파일 시스템을 나타냄.
 - `swap`, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 -(하이픈)이어야 합니다.

- 로컬 장치(미러)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다. 이들 키워드에 대한 설명은 77 페이지 “RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기”를 참조하십시오.

-n *BE_name*

생성될 부트 환경의 이름 *BE_name*은 고유해야 합니다.

새 부트 환경 만들기가 완료되면 업그레이드 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

예 4-6 부트 환경 만들기 및 공유 가능 파일 시스템 복사

이 예에서 현재 부트 환경은 루트(/)와 /home 등 두 파일 시스템을 포함합니다. 새 부트 환경에서 루트(/) 파일 시스템은 두 파일 시스템, 루트(/)와 /usr로 분리됩니다. /home 파일 시스템은 새 부트 환경으로 복사됩니다. 설명(mydescription)은 부트 환경 이름인 second_disk와 연결됩니다.

```
# lucreate -A 'mydescription' -c first_disk \
-m /:/dev/dsk/c0t4d0s0:ufs -m /usr:/dev/dsk/c0t4d0s3:ufs \
-m /home:/dev/dsk/c0t4d0s4:ufs -n second_disk
```

▼ 다른 소스에서 부트 환경 만들기

lucreate 명령을 활성 부트 환경의 파일 시스템을 기준으로 부트 환경을 작성합니다. 활성 부트 환경이 아닌 부트 환경을 기반으로 부트 환경을 만들려면 lucreate를 -s 옵션과 함께 사용합니다.

주 - 새 부트 환경을 활성화하고 폴백하려면 소스 부트 환경이 아닌 마지막 활성이었던 부트 환경으로 다시 부트합니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 부트 환경을 만듭니다.

```
# lucreate [-A 'BE_description'] -s source_BE_name
-m mountpoint:device[,metadevice]:fs_options -n BE_name
```

-A '*BE_description*'

(선택 사항) 부트 환경 이름(*BE_name*)과 연관된 부트 환경 설명을 작성할 수 있도록 합니다. 설명은 길이 제한이 없고 모든 문자를 포함할 수 있습니다.

`-s source_BE_name`

새 부트 환경에 대하여 소스 부트 환경을 지정합니다. 소스는 활성 부트 환경이 아닙니다.

`-m mountpoint:device[,metadevice]:fs_options [-m...]`

새 부트 환경의 파일 시스템의 구성을 지정합니다. `-m`에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- `mountpoint`는 모든 유효한 마운트 지점 또는 `-(하이픈)`일 수 있으며 스왑 분할 영역을 나타냅니다.
- `device` 필드는 다음 중 하나일 수 있습니다.
 - `/dev/dsk/cwtxdysz` 형식의 디스크 장치 이름
 - `/dev/md/dsk/dnum` 형식의 Solaris 볼륨 관리자 메타 장치 이름
 - `/dev/vx/dsk/volume_name` 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 `merged`는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냄.
- `fs_options` 필드는 다음 중 하나일 수 있습니다.
 - `ufs`, UFS 파일 시스템을 나타냄.
 - `vxfs`, Veritas 파일 시스템을 나타냄.
 - `swap`, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 `-(하이픈)`이어야 합니다.
 - 로컬 장치(미러)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다. 이들 키워드에 대한 설명은 77 페이지 “RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기”를 참조하십시오.

`-n BE_name`

생성될 부트 환경의 이름 `BE_name`은 시스템에서 고유해야 합니다.

새 부트 환경 만들기가 완료되면 업그레이드 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

예 4-7 다른 소스에서 부트 환경 만들기

이 예에서 이름이 `third_disk`인 소스 부트 환경에 있는 루트(/) 파일 시스템을 기반으로 부트 환경을 만들었으며, `third_disk`는 활성 부트 환경이 아닙니다.

설명(`mydescription`)은 `second_disk`라는 새 부트 환경과 연관됩니다.

```
# lucreate -A 'mydescription' -s third_disk \
-m /:/dev/dsk/c0t4d0s0:ufs -n second_disk
```

▼ Solaris Flash 아카이브용 빈 부트 환경 만들기

lucreate 명령은 활성 부트 환경에서 파일 시스템을 기반으로 하는 부트 환경을 만듭니다. lucreate 명령을 -s 옵션과 함께 사용하면 lucreate는 빈 부트 환경을 빠르게 만듭니다. 지정된 파일 시스템용으로 슬라이스가 예비되지만 파일 시스템은 복사되지 않습니다. 부트 환경의 이름은 지정되지만 Solaris Flash 아카이브를 사용하여 설치될 때까지 실제로 만들어지지 않습니다. 아카이브를 사용하여 빈 부트 환경이 설치되면 파일 시스템은 예약된 슬라이스에 설치됩니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 빈 부트 환경을 만듭니다.

```
# lucreate -A 'BE_name' -s - \
-m mountpoint:device[,metadevice]:fs_options -n BE_name
-A 'BE_description'
```

(선택 사항) 부트 환경 이름(BE_name)과 연관된 부트 환경 설명을 작성할 수 있도록 합니다. 설명은 길이 제한이 없고 모든 문자를 포함할 수 있습니다.

-s -

빈 부트 환경의 생성을 지정합니다.

```
-m mountpoint:device[,metadevice]:fs_options [-m...]
```

새 부트 환경의 파일 시스템의 구성을 지정합니다. -m에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- *mountpoint*는 모든 유효한 마운트 지점 또는 -(하이픈)일 수 있으며 스왑 분할 영역을 나타냅니다.
- *device* 필드는 다음 중 하나일 수 있습니다.
 - /dev/dsk/cwtxdysz 형식의 디스크 장치 이름
 - /dev/md/dsk/dnum 형식의 Solaris 볼륨 관리자 메타 장치 이름
 - /dev/vx/dsk/volume_name 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 merged는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냅니다.
- *fs_options* 필드는 다음 중 하나일 수 있습니다.
 - ufs, UFS 파일 시스템을 나타냅니다.
 - vxfs, Veritas 파일 시스템을 나타냅니다.
 - swap, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 -(하이픈)이어야 합니다.

- 로컬 장치(미러)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다. 이들 키워드에 대한 설명은 77 페이지 “RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기”을 참조하십시오.

-n *BE_name*

생성될 부트 환경의 이름 *BE_name*은 시스템에서 고유해야 합니다.

예 4-8 Solaris Flash 아카이브용 빈 부트 환경 만들기

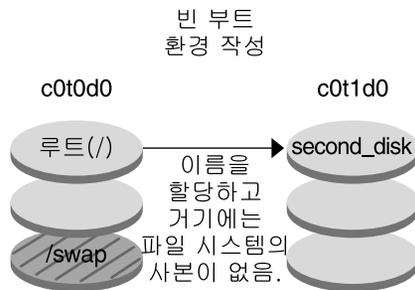
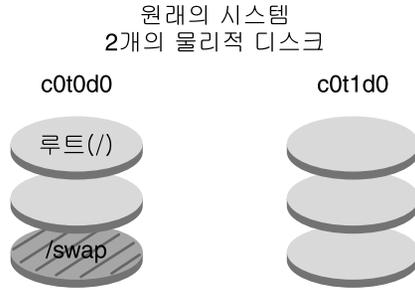
이 예에서 부트 환경이 생성되지만 파일 시스템을 포함하지는 않습니다. 설명(mydescription)은 second_disk라는 새 부트 환경과 연결됩니다.

```
# lucreate -A 'mydescription' -s - \
-m /:/dev/dsk/c0t1d0s0:ufs -n second_disk
```

빈 부트 환경의 만들기가 완료되면 플래시 아카이브를 설치하고 활성화(부트 가능)할 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

빈 부트 환경 만들기 및 채우기 예는 165 페이지 “빈 부트 환경 만들기 및 Solaris Flash 아카이브 설치 예”를 참조하십시오.

다음 이미지는 빈 부트 환경을 만드는 것입니다.



```
Command: # lucreate
        -m /:/dev/dsk/c0t1d0s0:ufs \
        -n second_disk
```

▼ RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기

부트 환경을 만들 때 Solaris Live Upgrade는 Solaris Volume Manager 기술을 사용하여 RAID-1 볼륨을 만듭니다. 부트 환경을 만들 때 Solaris Live Upgrade를 사용하여 다음 작업을 관리할 수 있습니다.

- RAID-1 볼륨(미러)에서 단일 슬라이스 연결(하위미러)을 제거합니다. 필요할 경우 새 부트 환경의 내용이 되도록 내용을 저장할 수 있습니다. 내용은 복사되지 않으므로 새 부트 환경을 빨리 만들 수 있습니다. 미러에서 분리된 하위미러는 더 이상 원래 미러에 속하지 않습니다. 하위미러에 대한 읽기 및 쓰기 작업은 더 이상 미러를 통해 수행되지 않습니다.
- 미러를 포함하는 부트 환경을 만듭니다.
- 단일 슬라이스 연결을 새로 만든 미러에 연결합니다.

시작하기 전에 Solaris Live Upgrade의 미러링 기능을 사용하여 상태 데이터베이스와 상태 데이터베이스 복제본을 만들어야 합니다. Solaris 볼륨 관리자 구성의 상태에 대한 디스크 정보를 저장하는 상태 데이터베이스입니다.

- 상태 데이터베이스 작성에 대한 자세한 내용은 **Solaris Volume Manager Administration Guide**의 6 장, “State Database (Overview)”를 참조하십시오.

- Solaris 볼륨 관리자 개요와 Solaris Live Upgrade가 제공할 수 있는 작업에 대한 개요는 27 페이지 “RAID-1 볼륨 파일 시스템이 있는 부트 환경 생성”을 참조하십시오.
- Solaris Live Upgrade를 사용할 때 허용되지 않는 복잡한 Solaris Volume Manager 구성에 대한 자세한 내용은 **Solaris Volume Manager Administration Guide**의 2 장, “Storage Management Concepts”를 참조하십시오.

1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 부트 환경을 만들려면 다음을 입력합니다.

```
# lucreate [-A 'BE_description'] \
-m mountpoint:device[,metadevice]:fs_options [-m...] \
-n BE_name
```

-A 'BE_description'

(선택 사항) 부트 환경 이름 *BE_name*과 연결된 부트 환경 설명을 만들 수 있습니다. 설명은 길이 제한이 없고 모든 문자를 포함할 수 있습니다.

-m mountpoint:device[,metadevice]:fs_options [-m...]

*vfstab*에 새 부트 환경의 파일 시스템 구성을 지정합니다. *-m*에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- *mountpoint*는 모든 유효한 마운트 지점 또는 -(하이픈)일 수 있으며 스왑 분할 영역을 나타냅니다.
- *device* 필드는 다음 중 하나일 수 있습니다.
 - /dev/dsk/cwtxdysz 형식의 디스크 장치 이름
 - /dev/md/dsk/dnum 형식의 Solaris 볼륨 관리자 볼륨 이름
 - /dev/md/vxfs/dsk/dnum 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 *merged*는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냄.
- *fs_options* 필드는 다음 파일 시스템 유형 및 키워드 중 하나입니다.
 - *ufs*, UFS 파일 시스템을 나타냄.
 - *vxfs*, Veritas 파일 시스템을 나타냄.
 - *swap*, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 -(하이픈)이어야 합니다.
 - 로컬 장치(미러)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다.

- **미러** 지정된 장치에 RAID-1 볼륨 또는 미러를 만듭니다. -m 옵션 다음에 **attach**를 지정하여 최소 하나의 연결을 새 미러에 추가해야 합니다. 지정된 장치 이름을 정확하게 지정해야 합니다. 예를 들어, /dev/md/dsk/d10의 논리적 장치 이름을 미러 이름으로 사용할 수 있습니다. 장치 이름 지정에 대한 자세한 내용은 **Solaris Volume Manager Administration Guide**의 “Overview of Solaris Volume Manager Components”를 참조하십시오.
- **detach**는 지정된 마운트 지점과 연결된 볼륨에서 연결을 제거합니다. 볼륨을 지정할 필요는 없습니다.
- **attach**는 지정된 마운트 지점과 연결된 미러에 연결을 추가합니다. 지정된 물리적 디스크 슬라이스는 미러에 추가할 단일 장치 연결이 됩니다. 디스크에 추가할 연결을 지정하려면 장치 이름에 쉼표와 해당 연결 이름을 추가합니다. 쉼표와 연결 이름을 생략하면 **lucreate**는 해당 연결에 대해 사용할 가능한 볼륨을 선택합니다.

lucreate를 사용하면 단일의 물리적 슬라이스를 포함하는 연결만 작성할 수 있습니다. 쉼표를 사용하여 미러에 연결을 세 개까지 추가할 수 있습니다.

- **preserve**는 기존 파일 시스템과 해당 내용을 저장합니다. 이 키워드를 사용하면 소스 부트 환경의 내용을 복사하는 복사 과정을 생략할 수 있습니다. 내용을 저장하면 새 부트 환경을 빠르게 만들 수 있습니다. 특정 마운트 지점의 경우 하나의 물리적 장치에 대해서만 **preserve**를 사용할 수 있습니다. **preserve**를 사용하면 **lucreate**는 장치 내용이 지정된 파일 시스템에 적합한지 확인합니다. 이 검사는 제한적이며 적합성을 보장할 수 없습니다.

preserve 키워드는 물리적 슬라이스 및 Solaris 볼륨 관리자 볼륨과 함께 사용할 수 있습니다.

- 실제 슬라이스에 UFS 파일 시스템이 있는 경우 **preserve** 키워드를 사용하면 UFS 파일 시스템의 내용이 해당 슬라이스에 저장됩니다. -m 옵션의 다음 예에서 **preserve** 키워드는 루트(/) 파일 시스템에 대한 마운트 지점의 파일 시스템으로 물리적 장치 **c0t0d0s0**의 내용을 저장합니다.

```
-m /:/dev/dsk/c0t0d0s0:preserve,ufs
```

- 볼륨에 UFS 파일 시스템이 있는 경우 **preserve** 키워드를 사용하면 UFS 파일 시스템의 내용이 해당 볼륨에 저장됩니다.

-m 옵션의 다음 예에서 **preserve** 키워드는 루트(/) 파일 시스템에 대한 마운트 지점의 파일 시스템으로 RAID-1 볼륨(미러) **d10**의 내용을 저장합니다.

```
-m /:/dev/md/dsk/d10:preserve,ufs
```

-m 옵션의 다음 예에서 RAID-1 볼륨(미러) **d10**은 루트(/) 파일 시스템에 대한 마운트 지점의 파일 시스템으로 구성됩니다. 단일 슬라이스 연결

d20은 현재 미러에서 분리됩니다. d20은 미러 d10에 연결됩니다. 루트(/) 파일 시스템은 하위미러 d20에 보관됩니다.

```
-m /:/dev/md/dsk/d10:mirror,ufs -m /:/dev/md/dsk/d20:detach,attach,preserve
```

```
-n BE_name
```

생성될 부트 환경의 이름 *BE_name*은 시스템에서 고유해야 합니다.

새 부트 환경 만들기가 완료되면 업그레이드 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

예 4-9 미러를 사용하여 부트 환경 만들기 및 장치 지정

이 예에서 파일 시스템의 마운트 지점은 -m 옵션을 사용하여 지정합니다.

- 설명(mydescription)은 another_disk에 연결됩니다.
- lucreate는 마운트 포인트 루트(/)에 대해 UFS 파일 시스템을 구성합니다. 미러 d10이 생성됩니다. 이 미러는 미러 d10에 복사된 현재 부트 환경의 루트(/) 파일 시스템에 대한 저장소입니다. 미러 d10에 있는 모든 데이터를 덮어씁니다.
- 두 슬라이스 c0t0d0s0과 c0t1d0s0은 하위미러인 d1과 d2입니다. 이 두 하위미러는 d10에 추가됩니다.
- 새 부트 환경의 이름은 another_disk로 지정됩니다.

```
# lucreate -A 'mydescription' \
-m /:/dev/md/dsk/d10:ufs,mirror \
-m /:/dev/dsk/c0t0d0s0,/dev/md/dsk/d1:attach \
-m /:/dev/dsk/c0t1c0s0,/dev/md/dsk/d2:attach -n another_disk
```

예 4-10 미러를 사용하여 부트 환경 만들기 및 하위 미러 이름 지정 안 함

이 예에서 파일 시스템의 마운트 지점은 -m 옵션을 사용하여 지정합니다.

- 설명(mydescription)은 another_disk에 연결됩니다.
- lucreate는 마운트 포인트 루트(/)에 대해 UFS 파일 시스템을 구성합니다. 미러 d10이 생성됩니다. 이 미러는 미러 d10에 복사된 현재 부트 환경의 루트(/) 파일 시스템에 대한 저장소입니다. 미러 d10에 있는 모든 데이터를 덮어씁니다.
- 두 슬라이스 c0t0d0s0과 c0t1d0s0은 하위미러로 사용되도록 지정됩니다. 하위미러를 지정하는 대신 lucreate 명령은 사용 가능한 볼륨 이름 목록에서 이름을 선택합니다. 이 두 개의 하위미러는 미러 d10에 연결됩니다.
- 새 부트 환경의 이름은 another_disk로 지정됩니다.

```
# lucreate -A 'mydescription' \
-m /:/dev/md/dsk/d10:ufs,mirror \
-m /:/dev/dsk/c0t0d0s0:attach \
-m /:/dev/dsk/c0t1d0s0:attach -n another_disk
```

새 부트 환경 만들기가 완료되면 업그레йд 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레йд(작업)”를 참조하십시오.

예 4-11 부트 환경 만들기 및 하위 미러 분리

이 예에서 파일 시스템의 마운트 지점은 -m 옵션을 사용하여 지정합니다.

- 설명(mydescription)은 another_disk에 연결됩니다.
- lucreate는 마운트 포인트 루트(/)에 대해 UFS 파일 시스템을 구성합니다. 미러 d10이 생성됩니다.
- 슬라이스 c0t0d0s0이 현재 미러에서 제거됩니다. 해당 슬라이스가 하위 미러 d1로 지정되고 미러 d10에 추가됩니다. 하위미러의 내용인 루트(/) 파일 시스템이 저장되며, 복사는 수행되지 않습니다. 슬라이스 c0t1d0s0은 하위미러 d2로서 미러 d10에 추가됩니다.
- 새 부트 환경의 이름은 another_disk로 지정됩니다.

```
# lucreate -A 'mydescription' \
-m /:/dev/md/dsk/d10:ufs,mirror \
-m /:/dev/dsk/c0t0d0s0,/dev/md/dsk/d1:detach,attach,preserve \
-m /:/dev/dsk/c0t1d0s0,/dev/md/dsk/d2:attach -n another_disk
```

새 부트 환경 만들기가 완료되면 업그레йд 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레йд(작업)”를 참조하십시오.

예 4-12 부트 환경 만들기, 하위 미러 분리, 내용 저장

이 예에서 파일 시스템의 마운트 지점은 -m 옵션을 사용하여 지정합니다.

- 설명(mydescription)은 another_disk에 연결됩니다.
- lucreate는 마운트 포인트 루트(/)에 대해 UFS 파일 시스템을 구성합니다. 미러 d20이 생성됩니다.
- 슬라이스 c0t0d0s0은 현재 미러에서 제거되었으며 미러 d20에 추가됩니다. 하위미러의 이름은 지정되지 않습니다. 하위미러의 내용인 루트(/) 파일 시스템이 저장되며, 복사는 수행되지 않습니다.
- 새 부트 환경의 이름은 another_disk로 지정됩니다.

```
# lucreate -A 'mydescription' \
-m /:/dev/md/dsk/d20:ufs,mirror \
-m /:/dev/dsk/c0t0d0s0:detach,attach,preserve \
-n another_disk
```

새 부트 환경의 만들기를 완료하면 부트 환경은 업그레йд되고 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레йд(작업)”를 참조하십시오.

예 4-13 두 개의 미러를 사용하여 부트 환경 만들기

이 예에서 파일 시스템의 마운트 지점은 `-m` 옵션을 사용하여 지정합니다.

- 설명(mydescription)은 `another_disk`에 연결됩니다.
- `lucreate`는 마운트 포인트 루트(/)에 대해 UFS 파일 시스템을 구성합니다. 미러 `d10`이 생성됩니다. 이 미러는 미러 `d10`에 복사된 현재 부트 환경의 루트(/) 파일 시스템에 대한 저장소입니다. 미러 `d10`에 있는 모든 데이터를 덮어씁니다.
- 두 슬라이스 `c0t0d0s0` 및 `c0t1d0s0`은 `d1` 및 `d2`의 하위미러입니다. 이 두 하위미러는 `d10`에 추가됩니다.
- `lucreate`는 마운트 지점 `/opt`에 대해 UFS 파일 시스템을 구성합니다. 미러 `d11`이 생성됩니다. 이 미러는 미러 `d11`에 복사된 현재 부트 환경의 `/opt` 파일 시스템에 대한 저장소입니다. 미러 `d11`에 있는 모든 데이터를 덮어씁니다.
- 두 슬라이스 `c2t0d0s1` 및 `c3t1d0s1`은 `d3` 및 `d4`의 하위미러입니다. 이 두 하위미러는 미러 `d11`에 추가됩니다.
- 새 부트 환경의 이름은 `another_disk`로 지정됩니다.

```
# lucreate -A 'mydescription' \
-m /:/dev/md/dsk/d10:ufs,mirror \
-m /:/dev/dsk/c0t0d0s0,/dev/md/dsk/d1:attach \
-m /:/dev/dsk/c0t1d0s0,/dev/md/dsk/d2:attach \
-m /opt:/dev/md/dsk/d11:ufs,mirror \
-m /opt:/dev/dsk/c2t0d0s1,/dev/md/dsk/d3:attach \
-m /opt:/dev/dsk/c3t1d0s1,/dev/md/dsk/d4:attach -n another_disk
```

새 부트 환경 만들기가 완료되면 업그레이드 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

▼ 부트 환경 만들기 및 내용 사용자 정의

새 부트 환경의 파일 시스템 내용은 다음 옵션을 사용하여 수정할 수 있습니다. 디렉토리 및 파일은 새 부트 환경에 복사되지 않습니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 부트 환경을 만들려면 다음을 입력합니다.

```
# lucreate -m mountpoint:device[,metadevice]:fs_options [-m ...] \
[-x exclude_dir] [-y include] \
[-Y include_list_file] \
[-f exclude_list_file]\
[-z filter_list] [-I] -n BE_name
```

-m *mountpoint:device[,metadevice]:fs_options* [-m...]

*vfstab*에 새 부트 환경의 파일 시스템 구성을 지정합니다. *-m*에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- *mountpoint*는 모든 유효한 마운트 지점 또는 *-(하이픈)*일 수 있으며 스왑 분할 영역을 나타냅니다.
- *device* 필드는 다음 중 하나일 수 있습니다.
 - */dev/dsk/cwtxdysz* 형식의 디스크 장치 이름
 - */dev/md/dsk/dnum* 형식의 Solaris 볼륨 관리자 볼륨 이름
 - */dev/md/vxfs/dsk/dnum* 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 *merged*는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냄.
- *fs_options* 필드는 다음 중 하나일 수 있습니다.
 - *ufs*, UFS 파일 시스템을 나타냄.
 - *vxfs*, Veritas 파일 시스템을 나타냄.
 - *swap*, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 *-(하이픈)*이어야 합니다.
 - 로컬 장치(미러)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다. 이들 키워드에 대한 설명은 77 페이지 “RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기”을 참조하십시오.

-x *exclude_dir*

새 부트 환경에 파일과 디렉토리를 복사하지 않고 제외합니다. 이 옵션의 여러 인스턴스를 사용하여 하나 이상의 파일 또는 디렉토리를 제외할 수 있습니다.

*exclude_dir*은 디렉토리 또는 파일의 이름입니다.

-y *include_dir*

새 부트 환경에 나열된 디렉토리 및 파일을 복사합니다. 디렉토리를 제외했지만 개별 하위 디렉토리 또는 파일을 복원하려 할 경우 이 옵션이 사용됩니다.

*include_dir*은 포함될 하위 디렉토리 또는 파일의 이름입니다.

-Y *list_filename*

목록의 디렉토리 및 파일을 새 부트 환경에 복사합니다. 디렉토리를 제외했지만 개별 하위 디렉토리 또는 파일을 복원하려 할 경우 이 옵션이 사용됩니다.

- *list_filename*은 목록을 포함하는 파일의 전체 경로입니다.
- *list_filename* 파일은 해당 하나의 파일을 포함해야 합니다.
- 행 항목이 디렉토리이면 해당 디렉토리 아래에 있는 모든 하위 디렉토리 및 파일이 포함됩니다. 행 항목이 파일이면 해당 파일만 포함됩니다.

-f list_filename

목록을 사용하여 디렉토리와 파일을 새 부트 환경에 복사하지 않고 제외합니다.

- list_filename은 목록을 포함하는 파일의 전체 경로입니다.
- list_filename 파일은 해당 하나의 파일을 포함해야 합니다.

-z list_filename

목록을 사용하여 디렉토리와 파일을 새 부트 환경에 복사합니다. 목록의 각 파일 또는 디렉토리는 플러스 “+” 또는 마이너스 “-”로 표시됩니다. 플러스는 포함된 파일 또는 디렉토리를 나타내고 마이너스는 제외된 파일 또는 디렉토리를 나타냅니다.

- list_filename은 목록을 포함하는 파일의 전체 경로입니다.
- list_filename 파일은 해당 하나의 파일을 포함해야 합니다. 파일 이름과 플러스 또는 마이너스 기호 사이에는 공백이 있어야 합니다.
- 행 항목이 디렉토리이고 +(플러스)로 지정된 경우 해당 디렉토리 아래에 있는 모든 하위 디렉토리와 파일이 포함됩니다. 행 항목이 파일이고 +(플러스)로 지정된 경우 해당 파일만 포함됩니다.

-I

시스템 파일의 무결성 검사를 무시합니다. 이 옵션은 주의해서 사용해야 합니다.

부트 환경에서 중요한 시스템 파일이 제거되지 않도록 하기 위해 lucreate는 무결성 검사를 실행합니다. 이 검사는 시스템 패키지 데이터베이스에 등록된 모든 파일들을 검사하며, 파일이 하나라도 제외되면 부트 환경 만들기를 중단합니다. 위의 옵션을 사용하면 이 무결성 검사가 무시됩니다. 이 옵션은 부트 환경을 빨리 만들기는 하지만 문제가 검색되지 않을 수 있습니다.

-n BE_name

생성될 부트 환경의 이름 BE_name은 시스템에서 고유해야 합니다.

새 부트 환경 만들기가 완료되면 업그레이드 및 활성화(부트 가능)될 수 있습니다. 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

예 4-14 부트 환경 만들기 및 파일 제외

이 예에서 새 부트 환경에는 second_disk라는 이름이 지정됩니다. 소스 부트 환경에는 하나의 파일 시스템, 루트(/)가 포함됩니다. 새 부트 환경에서 /var 파일 시스템은 루트(/) 파일 시스템에서 분리되어 다른 슬라이에 배치됩니다. lucreate 명령은 마운트 지점 루트(/) 및 /var에 대해 UFS 파일 시스템을 구성합니다. 또한 두 개의 /var 우편 파일, root 및 staff는 새 부트 환경에 복사되지 않습니다. 스왑은 원본 부트 환경과 새 부트 환경 간에 자동으로 공유됩니다.

```
# lucreate -n second_disk \
-m /:/dev/dsk/c0t1d0s0:ufs -m /var/mail:/dev/dsk/c0t2d0s0:ufs \
-x /var/mail/root -x /var/mail/staff
```

예 4-15 부트 환경 만들기 및 파일 제외/포함

이 예에서 새 부트 환경에는 `second_disk`라는 이름이 지정됩니다. 소스 부트 환경에는 OS에 대한 하나의 파일 시스템, 루트(/)가 포함됩니다. 소스에는 이름이 `/mystuff`인 파일 시스템도 포함됩니다. `lucreate`는 마운트 지점 루트(/) 및 `/mystuff`에 대해 UFS 파일 시스템을 구성합니다. `/mystuff`의 두 디렉토리만 새 부트 환경 `/latest` 및 `/backup`으로 복사됩니다. 스왑은 원본 부트 환경과 새 부트 환경 간에 자동으로 공유됩니다.

```
# lucreate -n second_disk \  
-m /:/dev/dsk/c0t1d0s0:ufs -m /mystuff:/dev/dsk/c1t1d0s0:ufs \  
-x /mystuff -y /mystuff/latest -y /mystuff/backup
```


Solaris Live Upgrade를 통해 업그레이드(작업)

이 장에서는 Solaris Live Upgrade를 사용하여 업그레이드하고 비활성 부트 환경을 활성화하는 방법에 대해 설명합니다.

주 - 이 장에서는 UFS 파일 시스템용 Solaris Live Upgrade에 대해 설명합니다. 사용법은 ZFS 부트 환경에서의 `luupgrade` 및 `luactivate` 명령과 동일합니다. UFS 파일 시스템을 ZFS 루트 풀로 마이그레이션하거나 ZFS 루트 풀을 만들고 설치하는 절차에 대한 자세한 내용은 13 장, “ZFS 루트 풀의 부트 환경 만들기”를 참조하십시오.

이 장은 다음 절로 구성되어 있습니다.

- 87 페이지 “작업 맵: 부트 환경 업그레이드”
- 88 페이지 “부트 환경 업그레이드”
- 102 페이지 “부트 환경에 Solaris Flash 아카이브 설치”
- 107 페이지 “부트 환경 활성화”

작업 맵: 부트 환경 업그레이드

표 5-1 작업: Solaris Live Upgrade를 사용하여 업그레이드

작업	설명	수행 방법
부트 환경을 업그레이드하거나 Solaris Flash 아카이브를 설치합니다.	<ul style="list-style-type: none"> ■ OS 이미지를 사용하여 비활성 부트 환경을 업그레이드합니다. ■ 비활성 부트 환경에 Solaris Flash 아카이브를 설치합니다. 	<ul style="list-style-type: none"> ■ 88 페이지 “부트 환경 업그레이드” ■ 102 페이지 “부트 환경에 Solaris Flash 아카이브 설치”

표 5-1 작업: Solaris Live Upgrade를 사용하여 업그레이드 (계속)

작업	설명	수행 방법
비활성 부트 환경을 활성화합니다.	변경 사항을 적용하고 비활성 부트 환경을 활성으로 전환합니다.	107 페이지 “부트 환경 활성화”
(선택 사항) 활성화할 때 장애가 발생하면 다시 원래 부트 환경으로 전환합니다.	장애가 발생하면 원래 부트 환경으로 다시 활성화합니다.	6 장, “오류 복구: 원래 부트 환경으로 롤백(작업)”

부트 환경 업그레이드

luupgrade 명령을 사용하여 부트 환경을 업그레이드합니다. 여기에서는 다음 매체에 있는 파일에서 비활성 부트 환경을 업그레이드하는 절차에 대하여 설명합니다.

- NFS 서버
- 로컬 파일
- 로컬 테이프
- DVD나 CD를 포함한 로컬 장치

업그레이드 지침

부트 환경을 최신 OS로 업그레이드하는 경우 활성 부트 환경에 영향을 주지 않습니다. 새 파일이 비활성 부트 환경 핵심 파일 시스템과 통합되지만 공유 가능 파일 시스템은 변경되지 않습니다.

RAID-1 볼륨이 설치되어 있거나 비전역 영역이 설치되어 있을 때 업그레이드하거나, Solaris Flash를 설치할 수 있습니다.

- 물리적 디스크 슬라이스, Solaris 볼륨 관리자 볼륨 또는 Veritas 볼륨 관리자 볼륨의 모든 조합을 포함하고 있는 비활성 부트 환경을 업그레이드할 수 있습니다. 루트(/) 파일 시스템에 대해 선택된 슬라이스는 RAID-1 볼륨(미러)에 포함된 단일 슬라이스 연결이어야 합니다. 미러된 파일 시스템으로 부트 환경을 만드는 절차에 대해서는 77 페이지 “RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기”를 참조하십시오.

주 - 현재 시스템에 VxVM 볼륨이 구성되어 있으면 lucreate 명령을 사용하여 새 부트 환경을 만들 수 있습니다. 데이터를 새 부트 환경으로 복사하면 Veritas 파일 시스템 구성이 손실되며 새 부트 환경에서 UFS 파일 시스템이 작성됩니다.

- 모든 설치 프로그램으로 비전역 영역이 설치된 시스템을 업그레이드할 수 있습니다. 비전역 영역이 설치된 경우 업그레이드하는 절차에 대해서는 8 장, “비전역 영역이 설치된 시스템에서 Solaris OS 업그레이드”를 참조하십시오.

- 업그레이드가 아니라 Solaris Flash 아카이브를 만든 경우 비활성 부트 환경에 아카이브를 설치할 수 있습니다. 새 파일은 비활성 부트 환경의 중요 파일 시스템을 덮어쓰지만 공유 가능한 파일 시스템은 변경되지 않습니다. 102 페이지 “부트 환경에 Solaris Flash 아카이브 설치”를 참조하십시오.

패키지 또는 패치로 시스템 업그레이드

Solaris Live Upgrade를 사용하여 시스템에 패치 및 패키지를 추가할 수 있습니다. Solaris Live Upgrade는 현재 실행 중인 시스템의 복사본을 만듭니다. 새 부트 환경을 업그레이드하거나 패키지 또는 패치를 추가할 수 있습니다. Solaris Live Upgrade를 사용하는 경우 시스템을 다시 부트하는 경우에만 작동이 중지됩니다. `luupgrade` 명령을 사용하여 새 부트 환경에 패치와 패키지를 추가할 수 있습니다.



주의 - 패키지나 패치를 추가 및 제거하는 경우 Solaris Live Upgrade에 SVR4 고급 패키징 지침을 준수하는 패키지 또는 패치가 필요합니다. Sun 패키지가 이러한 지침을 준수하는 반면 Sun은 타사 공급업체의 패키지 준수를 보장할 수 없습니다. 패키지가 이러한 지침을 위반하면 업그레이드하는 동안 패키지 추가 소프트웨어에 오류가 발생하거나 활성 부트 환경이 변경됩니다.

패키지 요구 사항에 대한 자세한 내용은 부록 B, “추가 SVR4 패키지 요구 사항(참조)”을 참조하십시오.

표 5-2 패키지 및 패치를 사용하여 부트 환경 업그레이드

설치 유형	설명	자세한 정보
부트 환경에 패치 추가	새 부트 환경을 만들고 <code>luupgrade</code> 명령을 <code>-t</code> 옵션과 함께 사용합니다.	93 페이지 “부트 환경에서 네트워크 설치 이미지에 패치 추가”
부트 환경에 패키지 추가	<code>luupgrade</code> 명령을 <code>-p</code> 옵션과 함께 사용합니다.	92 페이지 “부트 환경에 네트워크 설치 이미지에 패키지 추가”

▼ 부트 환경에서 네트워크 설치 이미지 업그레이드

이 절차를 사용하여 업그레이드하려면 DVD 또는 네트워크 설치 이미지를 사용해야 합니다. 설치에 둘 이상의 CD가 필요한 경우 90 페이지 “여러 CD를 사용하여 네트워크 설치 이미지 업그레이드” 절차를 사용해야 합니다.

- 1 **Solaris Live Upgrade** `SUNWlucfg`, `SUNWlur` 및 `SUNWluu` 패키지를 시스템에 설치합니다. 이 패키지는 업그레이드하는 릴리스에서 가져온 것이어야 합니다. 단계별 절차는 58 페이지 “`pkgadd` 명령을 사용하여 Solaris Live Upgrade 설치”를 참조하십시오.

2 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

3 다음을 입력하여 업그레이드할 부트 환경과 설치 소프트웨어의 경로를 나타냅니다.

```
# luupgrade -u -n BE_name -s os_image_path
```

-u 부트 환경에서 네트워크 설치 이미지 업그레이드

-n BE_name 업그레이드될 부트 환경의 이름을 지정합니다.

-s os_image_path 네트워크 설치 이미지가 포함된 디렉토리의 경로 이름을 지정합니다.

예 5-1 DVD 매체를 사용하여 부트 환경의 네트워크 설치 이미지 업그레이드

이 예에서 second_disk 부트 환경은 DVD 매체를 사용하여 업그레이드됩니다. pkgadd 명령은 업그레이드하는 릴리스의 Solaris Live Upgrade 패키지를 추가합니다.

```
# pkgadd -d /server/packages SUNWLucfg SUNWlur SUNWluu
# luupgrade -u -n second_disk -s /cdrom/cdrom0
```

예 5-2 네트워크 설치 이미지를 사용하여 부트 환경의 네트워크 설치 이미지 업그레이드

이 예에서는 second_disk 부트 환경이 업그레이드됩니다. pkgadd 명령은 업그레이드하는 릴리스의 Solaris Live Upgrade 패키지를 추가합니다.

```
# pkgadd -d /server/packages SUNWLucfg SUNWlur SUNWluu
# luupgrade -u -n second_disk \
-s /net/installmachine/export/Solaris_10/OS_image
```

▼ 여러 CD를 사용하여 네트워크 설치 이미지 업그레이드

네트워크 설치 이미지가 둘 이상의 CD에 있기 때문에 이 업그레이드 절차를 사용해야 합니다. luupgrade 명령을 -i 옵션과 함께 사용하여 모든 추가 CD를 설치합니다.

- 1 **Solaris Live Upgrade** SUNWLucfg, SUNWlur 및 SUNWluu 패키지를 시스템에 설치합니다. 이 패키지는 업그레이드하는 릴리스에서 가져온 것이어야 합니다. 단계별 절차는 [58 페이지 “pkgadd 명령을 사용하여 Solaris Live Upgrade 설치”](#)를 참조하십시오.

2 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

3 다음을 입력하여 업그레이드할 부트 환경과 설치 소프트웨어의 경로를 나타냅니다.

```
# luupgrade -u -n BE_name -s os_image_path
```

-u 부트 환경에서 네트워크 설치 이미지 업그레이드
 -n *BE_name* 업그레이드될 부트 환경의 이름을 지정합니다.
 -s *os_image_path* 네트워크 설치 이미지가 포함된 디렉토리의 경로 이름을 지정합니다.

4 설치 프로그램이 첫 번째 CD의 내용을 완료하면 두 번째 CD를 삽입합니다.

5 이 단계는 이전 단계와 동일하지만 -u 옵션이 -i 옵션으로 교체됩니다. 또한 메뉴 또는 텍스트로 두 번째 CD의 설치 프로그램을 실행합니다.

- 이 명령은 메뉴로 두 번째 CD에서 설치 프로그램을 실행합니다.

```
# luupgrade -i -n BE_name -s os_image_path
```

- 이 명령은 두 번째 CD에서 텍스트로 설치 프로그램을 실행하며 사용자 상호 작용이 필요 없습니다.

```
# luupgrade -i -n BE_name -s os_image_path -O '-nodisplay -noconsole'
```

-i 추가 CD 설치. 소프트웨어는 지정된 매체에서 설치 프로그램을 찾아 해당 프로그램을 실행합니다. 설치 프로그램은 -s로 지정됩니다.
 -n *BE_name* 업그레이드될 부트 환경의 이름을 지정합니다.
 -s *os_image_path* 네트워크 설치 이미지가 포함된 디렉토리의 경로 이름을 지정합니다.
 -O '-nodisplay -noconsole' (선택 사항) 텍스트 모드로 두 번째 CD에서 설치 프로그램을 실행하며 사용자 상호 작용은 필요하지 않습니다.

6 설치할 각 CD에 대해 단계 4 및 단계 5를 반복합니다.

부트 환경이 활성화될 준비가 되었습니다. 107 페이지 “부트 환경 활성화”를 참조하십시오.

예 5-3 SPARC: 여러 CD를 사용하여 네트워크 설치 이미지 업그레이드

이 예에서 `second_disk` 부트 환경이 업그레이드되며 설치 이미지는 Solaris Software - 1 및 Solaris Software - 2 CD에 있습니다. `-u` 옵션은 CD 세트에 있는 모든 패키지에 대하여 충분한 공간이 있는지 확인합니다. `-o` 옵션과 함께 `-nodisplay` 및 `-noconsole` 옵션을 사용하면 두 번째 CD를 읽은 후 문자 사용자 인터페이스가 표시되지 않습니다. 이 옵션을 사용하면 정보를 입력하라는 프롬프트가 표시되지 않습니다.

참고: `-o` 옵션을 `-nodisplay` 및 `-noconsole` 옵션과 함께 사용하지 않으면 문자 사용자 인터페이스(CUI)가 표시됩니다. Sun에서는 Solaris Live Upgrade 작업 수행 시 CUI의 사용을 더 이상 권장하지 않습니다.

업그레이드하는 릴리스의 Solaris Live Upgrade 패키지를 설치합니다.

```
# pkgadd -d /server/packages SUNWLucfg SUNWlur SUNWluu
```

Solaris Software - 1 CD를 삽입하고 다음을 입력합니다.

```
# luupgrade -u -n second_disk -s /cdrom/cdrom0/
```

Solaris Software - 2 CD를 삽입하고 다음을 입력합니다.

```
# luupgrade -i -n second_disk -s /cdrom/cdrom0 -o '-nodisplay \
-noconsole'
```

Repeat this step for each CD that you need.

설치할 각 CD에 대해 앞의 단계를 반복합니다.

▼ 부트 환경에 네트워크 설치 이미지에 패키지 추가

다음은 새 부트 환경에서 패키지를 제거하고 추가하는 절차입니다.



주의 - 업그레이드하는 경우, 패키지나 패치를 추가 및 제거하는 경우 Solaris Live Upgrade에 SVR4 고급 패키징 지침을 준수하는 패키지 또는 패치가 필요합니다. Sun 패키지가 이러한 지침을 준수하는 반면 Sun은 타사 공급업체의 패키지 준수를 보장할 수 없습니다. 패키지가 이러한 지침을 위반하면 업그레이드하는 동안 패키지 추가 소프트웨어에 오류가 발생하거나 활성 부트 환경이 변경됩니다.

패키지 요구 사항에 대한 자세한 내용은 부록 B, “추가 SVR4 패키지 요구 사항(참조)”을 참조하십시오.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 새 부트 환경에서 패키지 또는 패키지 집합을 제거하려면 다음을 입력합니다.

```
# luupgrade -P -n second_disk package-name
```

-P 명명된 패키지를 부트 환경에서 제거할 것을 나타냅니다.

-n *BE_name* 패키지를 제거할 부트 환경의 이름을 지정합니다.

package-name 제거할 패키지의 이름을 지정합니다. 공백을 사용하여 여러 패키지 이름을 구분합니다.

3 새 부트 환경에 패키지 또는 패키지 집합을 추가하려면 다음을 입력합니다.

```
# luupgrade -p -n second_disk -s /path-to-packages package-name
```

-p 부트 환경에 패키지를 추가할 것을 나타냅니다.

-n *BE_name* 패키지를 추가할 부트 환경의 이름을 지정합니다.

-s *path-to-packages* 추가할 패키지가 포함된 디렉토리의 경로를 지정합니다.

package-name 추가할 패키지의 이름을 지정합니다. 공백을 사용하여 여러 패키지 이름을 구분합니다.

예 5-4 부트 환경에서 네트워크 설치 이미지에 패키지 추가

이 예에서 패키지는 제거된 다음 *second_disk* 부트 환경에 추가됩니다.

```
# luupgrade -P -n second_disk SUNWabc SUNWdef SUNWghi
# luupgrade -p -n second_disk -s /net/installmachine/export/packages \
SUNWijk SUNWlmn SUNWpkr
```

▼ 부트 환경에서 네트워크 설치 이미지에 패치 추가

다음은 새 부트 환경에서 패치를 제거하고 추가하는 절차입니다.



주의 - 패키지 또는 패치를 추가하고 제거할 때, Solaris Live Upgrade에는 SVR4 고급 패키징 작성 지침을 준수하는 패키지 또는 패치가 필요합니다. Sun 패키지가 이러한 지침을 준수하는 반면 Sun은 타사 공급업체의 패키지 준수를 보장할 수 없습니다. 패키지가 이러한 지침을 위반하면 업그레이드하는 동안 패키지 추가 소프트웨어에 오류가 발생하거나 활성 부트 환경이 변경됩니다.



주의 - 활성 부트 환경에서 Solaris 8 또는 9 OS를 실행 중일 때 Solaris Live Upgrade를 사용하여 Solaris 10 비활성 부트 환경을 패치할 수 없습니다. Solaris Live Upgrade는 활성 부트 분할 영역에서 패치 유틸리티를 호출하여 비활성 부트 분할 영역을 패치합니다. Solaris 8 및 Solaris 9 패치 유틸리티는 Solaris 10 OS의 Solaris 영역, 서비스 관리 기능(Service Management Facility, SMF) 및 기타 향상된 기능을 인식하지 못합니다. 따라서 패치 유틸리티는 비활성 Solaris 10 부트 환경을 제대로 패치하지 못합니다. 그러므로 Solaris Live Upgrade를 사용하여 시스템을 Solaris 8 또는 Solaris 9 OS에서 Solaris 10 OS로 업그레이드할 경우 패치를 실행하기 전에 Solaris 10 부트 환경을 활성화해야 합니다. Solaris 10 부트 환경이 활성화된 이후에 활성 부트 환경을 직접 패치하거나 다른 비활성 부트 환경을 설정하고 Solaris Live Upgrade를 사용하여 해당 환경을 패치할 수 있습니다. Solaris 8에서 Solaris 10 릴리스로의 업그레이드 및 패치에 대한 예는 [Solaris Live Upgrade 사용 제한](#)을 참조하십시오.

1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 새 부트 환경에서 패치 또는 패치 집합을 제거하려면 다음을 입력합니다.

```
# luupgrade -T -n second_disk patch_name
```

-T 명명된 패치를 부트 환경에서 제거할 것을 나타냅니다.

-n *BE_name* 패치를 제거할 부트 환경 이름을 지정합니다.

patch-name 제거할 패치 이름을 지정합니다. 공백을 사용하여 여러 패치 이름을 구분합니다.

3 새 부트 환경에 패치 또는 패치 집합을 추가하려면 다음 명령을 입력합니다.

```
# luupgrade -t -n second_disk -s /path-to-patches patch-name
```

-t 부트 환경에 패치를 추가할 것을 나타냅니다.

-n *BE_name* 패치를 추가할 부트 환경 이름을 지정합니다.

-s *path-to-patches* 추가할 패치가 포함된 디렉토리 경로를 지정합니다.

patch-name 추가할 패치 이름을 지정합니다. 공백을 사용하여 여러 패치 이름을 구분합니다.

예 5-5 부트 환경에서 네트워크 설치 이미지에 패치 추가

이 예에서 패치는 제거된 다음 `second_disk` 부트 환경에 추가됩니다.

```
# luupgrade -T -n second_disk 222222-01
# luupgrade -t -n second_disk -s /net/installmachine/export/packages \
333333-01 444444-01
```

▼ 부트 환경에 설치된 패키지에 대한 정보 가져오기

다음은 새 부트 환경에 설치된 패키지의 무결성을 검사하는 절차입니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 새 부트 환경에 새로 설치된 패키지의 무결성을 검사하려면 다음을 입력합니다.

```
# luupgrade -C -n BE_name -O "-v" package-name
```

-C 명명된 패키지에서 pkgchk 명령을 실행할 것을 나타냅니다.

-n BE_name 검사를 수행할 부트 환경 이름을 지정합니다.

-O 옵션을 pkgchk 명령에 직접 전달합니다.

package-name 검사할 패키지 이름을 지정합니다. 공백을 사용하여 여러 패키지 이름을 구분합니다. 패키지 이름을 생략하면 지정된 부트 환경의 모든 패키지에서 검사가 수행됩니다.

“-v” 상세 표시 모드에서 명령을 실행하도록 지정합니다.

예 5-6 부트 환경에서 패키지의 무결성 확인

이 예에서는 SUNWabc, SUNWdef 및 SUNWghi 패키지를 검사하여 패키지가 제대로 설치되었고 손상되지 않았는지 확인합니다.

```
# luupgrade -C -n second_disk SUNWabc SUNWdef SUNWghi
```

JumpStart 프로필을 사용하여 업그레이드

Solaris Live Upgrade에 사용할 JumpStart 프로필을 만들 수 있습니다. 사용자 정의 JumpStart 프로그램을 잘 알고 있으면 이 프로필은 사용자 정의 JumpStart에 사용되는 것과 같습니다. 다음 절차를 사용하면 프로필을 작성 및 테스트하고 luupgrade 명령을 -j 옵션과 함께 사용하여 프로필을 설치할 수 있습니다.



주의 - Solaris Flash 아카이브를 사용하여 Solaris OS를 설치하는 경우 아카이브와 설치 매체에 포함된 OS 버전이 같아야 합니다. 예를 들어, 아카이브가 Solaris 10 운영 체제이고 DVD 매체를 사용하는 경우 Solaris 10 DVD 매체를 사용하여 아카이브를 설치해야 합니다. OS 버전이 일치하지 않으면 대상 시스템에서 설치가 실패합니다. 다음 키워드나 명령을 사용할 때는 운영 체제가 같아야 합니다.

- 프로필의 `archive_location` 키워드
- `s`, `-a`, `-j` 및 `-J` 옵션과 함께 사용하는 `-luupgrade` 명령

자세한 내용은 다음을 참조하십시오.

- 96 페이지 “Solaris Live Upgrade에서 사용할 프로필 만들기”
- 100 페이지 “Solaris Live Upgrade에서 사용할 프로필 테스트”
- 101 페이지 “Solaris Live Upgrade를 사용하여 프로필로 업그레이드”
- JumpStart 프로필을 만들려면 **Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “프로필 작성”**을 참조하십시오.

▼ Solaris Live Upgrade에서 사용할 프로필 만들기

이 절차는 Solaris Live Upgrade와 함께 사용할 프로필을 작성하는 방법을 보여줍니다. `luupgrade` 명령을 `-j` 옵션과 함께 사용하여 비활성 부트 환경을 업그레이드하는 데 이 프로필을 사용할 수 있습니다.

이 프로필을 사용하는 절차는 다음 절을 참조하십시오.

- 프로필을 사용한 업그레이드에 대해서는 101 페이지 “Solaris Live Upgrade를 사용하여 프로필로 업그레이드”를 참조하십시오.
- 프로필을 사용한 Solaris Flash 설치에 대해서는 104 페이지 “프로필을 사용하여 Solaris Flash 아카이브 설치”를 참조하십시오.

1 텍스트 편집기를 사용하여 텍스트 파일을 만듭니다.

파일의 이름을 자세하게 지정합니다. 프로필을 사용하여 시스템에 Solaris 소프트웨어를 설치하는 방법을 프로필의 이름이 반영하도록 합니다. 예를 들어, 이 프로필의 이름을 `upgrade_Solaris_10`으로 지정할 수 있습니다.

2 프로필 키워드와 값을 프로필에 추가합니다.

Solaris Live Upgrade 프로필에는 다음 표에 있는 업그레이드 키워드만을 사용할 수 있습니다.

다음 표에서는 `upgrade` 또는 `flash_install`의 `Install_type` 키워드 값과 함께 사용할 수 있는 키워드를 나열합니다.

최초 아카이브 생성용 키워드	설명	참조
(필수) <code>Install_type</code>	<p>시스템의 기존 Solaris 환경을 업그레이드하거나 시스템에 Solaris Flash 아카이브를 설치할 것인지 여부를 정의합니다. 이 키워드와 함께 다음 값을 사용합니다.</p> <ul style="list-style-type: none"> ■ <code>upgrade</code> - 업그레이드의 경우 ■ <code>flash_install</code> - Solaris Flash 설치의 경우 ■ <code>flash_update</code> - Solaris Flash 차등 설치의 경우 	<p>이 키워드의 모든 값에 대한 설명은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “<code>install_type</code> 프로필 키워드(UFS 및 ZFS)”를 참조하십시오.</p>
(Solaris Flash 아카이브에 필수) <code>archive_location</code>	<p>지정된 위치에서 Solaris Flash 아카이브를 검색합니다.</p>	<p>이 키워드에 사용할 수 있는 값 목록은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “<code>archive_location</code> 키워드”를 참조하십시오.</p>
(선택 사항) <code>cluster</code> (클러스터 추가 또는 삭제)	<p>클러스터가 시스템에 설치될 소프트웨어 그룹에 추가 또는 삭제될 것인지 여부를 지정합니다.</p>	<p>이 키워드에 사용할 수 있는 값 목록은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “<code>cluster</code> 프로필 키워드(소프트웨어 그룹 추가)(UFS 및 ZFS)”를 참조하십시오.</p>
(선택 사항) <code>geo</code>	<p>시스템에 설치하거나 시스템을 업그레이드할 때 추가하고자 하는 지역 로케일을 지정합니다.</p>	<p>이 키워드에 사용할 수 있는 값 목록은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “<code>geo</code> 프로필 키워드(UFS 및 ZFS)”를 참조하십시오.</p>
(선택 사항) <code>local_customization</code>	<p>Solaris Flash 아카이브를 복제 시스템에 설치하기 전에 사용자 정의 스크립트를 만들어 복제 시스템에 로컬 구성을 보존할 수 있습니다. <code>local_customization</code> 키워드는 이러한 스크립트를 저장한 디렉토리를 지정합니다. 값은 복제 시스템에 있는 스크립트의 경로입니다.</p>	<p>배치 전 및 배치 후 스크립트에 대한 자세한 내용은 Solaris 10 5/09 설치 설명서: Solaris Flash 아카이브(작성 및 설치)의 “사용자 정의 스크립트 만들기”를 참조하십시오.</p>
(선택 사항) <code>locale</code>	<p>업그레이드 시 설치하거나 추가하고자 하는 로케일 패키지를 지정합니다.</p>	<p>이 키워드에 사용할 수 있는 값 목록은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “<code>locale</code> 프로필 키워드(UFS 및 ZFS)”를 참조하십시오.</p>
(선택 사항) <code>package</code>	<p>패키지가 시스템에 설치될 소프트웨어 그룹에 추가 또는 삭제될 것인지 여부를 지정합니다.</p>	<p>이 키워드에 사용할 수 있는 값 목록은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “<code>package</code> 프로필 키워드(UFS 및 ZFS)”를 참조하십시오.</p>

다음 표에는 `Install_type` 키워드 값 `flash_update`와 함께 사용할 수 있는 키워드가 나열됩니다.

차등 아카이브 생성용 키워드	설명	참조
(필수) <code>install_type</code>	시스템에 Solaris Flash 아카이브를 설치할 설치 경의 합니다. 차등 아카이브의 값은 <code>flash_update</code> 입니다.	이 키워드의 모든 값에 대한 설명은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “install_type 프로파일 키워드(UFS 및 ZFS)” 를 참조하십시오.
(필수) <code>archive_location</code>	지정된 위치에서 Solaris Flash 아카이브를 검색합니다.	이 키워드에 사용할 수 있는 값 목록은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “archive_location 키워드” 를 참조하십시오.
(선택 사항) <code>forced_deployment</code>	소프트웨어에 필요한 시스템과 다른 복제 시스템에 Solaris Flash 차등 아카이브를 강제 설치합니다. <code>forced_deployment</code> 를 사용하는 경우, 예상 상태로 복제 시스템을 가져오기 위해 새로운 모든 파일이 삭제됩니다. 파일의 삭제 여부를 결정하지 않은 경우, 설치를 중단하여 새 파일을 보호하는 기본값을 사용합니다.	이 키워드에 대한 자세한 내용은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “forced_deployment 프로파일 키워드(Solaris Flash 차등 아카이브 설치)” 를 참조하십시오.
(선택 사항) <code>local_customization</code>	복제 시스템에 Solaris Flash 아카이브를 설치하기 전에 먼저 사용자 정의 스크립트를 사용하여 복제 시스템에 로컬 구성을 유지할 수 있습니다. <code>local_customization</code> 키워드는 이러한 스크립트를 저장한 디렉토리를 지정합니다. 값은 복제 시스템에 있는 스크립트의 경로입니다.	배치 전 및 배치 후 스크립트에 대한 자세한 내용은 Solaris 10 5/09 설치 설명서: Solaris Flash 아카이브(작성 및 설치)의 “사용자 정의 스크립트 만들기” 를 참조하십시오.
(선택 사항) <code>no_content_check</code>	복제 시스템을 Solaris Flash 차등 아카이브와 함께 설치할 경우, <code>no_content_check</code> 키워드를 사용하여 파일별 검증을 무시할 수 있습니다. 파일별 검증은 복제 시스템이 마스터 시스템의 복제본임을 확인합니다. 복제 시스템이 원본 마스터 시스템의 복제본인지 확실하지 않으면 이 키워드를 사용하지 마십시오.	이 키워드에 대한 자세한 내용은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “no_content_check 프로파일 키워드(Solaris Flash 아카이브 설치)” 를 참조하십시오.
(선택 사항) <code>no_master_check</code>	복제 시스템을 Solaris Flash 차등 아카이브와 함께 설치할 경우 <code>no_master_check</code> 키워드를 사용하여 파일 확인을 무시할 수 있습니다. 복제 시스템 파일은 검사하지 않습니다. 검사는 복제가 원본 마스터 시스템에서 구축되었는지를 확인합니다. 복제 시스템이 원본 마스터 시스템의 복제본인지 확실하지 않으면 이 키워드를 사용하지 마십시오.	이 키워드에 대한 자세한 내용은 Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치의 “no_master_check 프로파일 키워드(Solaris Flash 아카이브 설치)” 를 참조하십시오.

3 로컬 시스템의 디렉토리에 프로파일을 저장합니다.

4 root가 프로파일을 소유하고 권한이 644로 설정되도록 합니다.

5 (선택 사항) 프로필을 테스트합니다.

프로필을 테스트하는 절차는 100 페이지 “Solaris Live Upgrade에서 사용할 프로필 테스트”를 참조하십시오.

예 5-7 Solaris Live Upgrade 프로필 만들기

이 예에서 프로필은 업그레이드 매개 변수를 제공합니다. 이 프로필은 Solaris Live Upgrade luupgrade 명령과 -u 및 -j 옵션으로 비활성 부트 환경을 업그레이드하는 데 사용됩니다. 이 프로필은 패키지 및 클러스터를 추가합니다. 지역 로캘 및 추가 로캘도 프로필에 추가됩니다. 프로필에 로캘을 추가할 경우 추가 디스크 공간으로 부트 환경을 만들었는지 확인합니다.

```
# profile keywords      profile values
# -----
install_type           upgrade
package                SUNWxman add
cluster                SUNWCacc add
geo                    C_Europe
locale                 zh_TW
locale                 zh_TW.BIG5
locale                 zh_TW.UTF-8
locale                 zh_HK.UTF-8
locale                 zh_HK.BIG5HK
locale                 zh
locale                 zh_CN.GB18030
locale                 zh_CN.GBK
locale                 zh_CN.UTF-8
```

예 5-8 차등 아카이브를 설치할 Solaris Live Upgrade 프로필 만들기

다음 프로필 예는 Solaris Live Upgrade에서 복제 시스템에 차등 아카이브를 설치할 때 사용됩니다. 차등 아카이브가 지정한 파일만 추가, 삭제 또는 변경됩니다. Solaris Flash 아카이브가 NFS 서버에서 검색됩니다. 이미지가 원본 마스터 시스템에서 구축되었기 때문에 복제 시스템이 유효한 시스템 이미지를 갖는지 확인되지 않습니다. 이 프로필은 Solaris Live Upgrade luupgrade 명령과 -u 및 -j 옵션과 함께 사용됩니다.

```
# profile keywords      profile values
# -----
install_type           flash_update
archive_location       nfs installserver:/export/solaris/archive/solarisarchive
no_master_check
```

luupgrade 명령을 사용하여 차등 아카이브를 설치하려면 104 페이지 “프로필을 사용하여 Solaris Flash 아카이브 설치”를 참조하십시오.

▼ Solaris Live Upgrade에서 사용할 프로필 테스트

프로필을 작성한 후 `luupgrade` 명령을 사용하여 프로필을 테스트하십시오. `luupgrade`에 의해 생성된 설치 출력을 조사하여 의도한 대로 프로필이 작동하는지 신속하게 확인할 수 있습니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 프로필을 테스트합니다.

```
# luupgrade -u -n BE_name -D -s os_image_path -j profile_path
```

-u	부트 환경에서 운영 체제 이미지를 업그레이드합니다.
-n BE_name	업그레이드될 부트 환경의 이름을 지정합니다.
-D	<code>luupgrade</code> 명령은 선택된 부트 환경의 디스크 구성을 사용하여 -j 옵션과 함께 전달되는 프로필 옵션을 테스트합니다.
-s os_image_path	운영 체제 이미지를 포함하는 디렉토리의 경로 이름을 지정합니다. 이 디렉토리는 DVD-ROM, CD-ROM과 같은 설치 매체에 있을 수도 있고 NFS 또는 UFS 디렉토리일 수도 있습니다.
-j profile_path	업그레이드용으로 구성된 프로필에 대한 경로입니다. 프로필은 로컬 시스템의 디렉토리에 있어야 합니다.

예 5-9 Solaris Live Upgrade를 사용하여 프로필 테스트

다음 예에서 프로필 이름은 `Flash_profile`입니다. 이 프로필은 `second_disk`라는 비활성 부트 환경에서 성공적으로 테스트됩니다.

```
# luupgrade -u -n u1b08 -D -s /net/installsvr/export/u1/combined.u1wos \
-j /var/tmp/flash_profile
Validating the contents of the media /net/installsvr/export/u1/combined.u1wos.
The media is a standard Solaris media.
The media contains an operating system upgrade image.
The media contains Solaris version 10.
Locating upgrade profile template to use.
Locating the operating system upgrade program.
Checking for existence of previously scheduled Live Upgrade requests.
Creating upgrade profile for BE second_disk.
Determining packages to install or upgrade for BE second_disk.
Simulating the operating system upgrade of the BE second_disk.
The operating system upgrade simulation is complete.
INFORMATION: var/sadm/system/data/upgrade_cleanup contains a log of the
```

upgrade operation.
 INFORMATION: var/sadm/system/data/upgrade_cleanup contains a log of
 cleanup operations required.
 The Solaris upgrade of the boot environment second_disk is complete.

이제 프로필을 사용하여 비활성 부트 환경을 업그레이드할 수 있습니다.

▼ Solaris Live Upgrade를 사용하여 프로필로 업그레이드

이 절차에서는 프로필을 사용하여 OS를 업그레이드하는 단계별 설명이 제공됩니다.

프로필을 사용하여 Solaris Flash 아카이브를 설치하려면 104 페이지 “프로필을 사용하여 Solaris Flash 아카이브 설치”를 참조하십시오.

프로필에 로케를 추가한 경우 추가 디스크 공간으로 부트 환경을 만들었는지 확인합니다.



주의 - Solaris Flash 아카이브를 사용하여 Solaris OS를 설치하는 경우 아카이브와 설치 매체에 포함된 OS 버전이 같아야 합니다. 예를 들어, 아카이브가 Solaris 10 운영 체제이고 DVD 매체를 사용하는 경우 Solaris 10 DVD 매체를 사용하여 아카이브를 설치해야 합니다. OS 버전이 일치하지 않으면 대상 시스템에서 설치가 실패합니다. 다음 키워드나 명령을 사용할 때는 운영 체제가 같아야 합니다.

- 프로필의 archive_location 키워드
- s, -a, -j 및 -J 옵션과 함께 사용하는 -luupgrade 명령

1 **Solaris Live Upgrade** SUNWlucfg, SUNWlur 및 SUNWluu 패키지를 시스템에 설치합니다. 이 패키지는 업그레이드하는 릴리스에서 가져온 것이어야 합니다. 단계별 절차는 58 페이지 “pkgadd 명령을 사용하여 Solaris Live Upgrade 설치”를 참조하십시오.

2 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

3 프로필을 만듭니다.

Solaris Live Upgrade 프로필에서 사용할 수 있는 업그레이드 키워드 목록은 96 페이지 “Solaris Live Upgrade에서 사용할 프로필 만들기”를 참조하십시오.

4 다음을 입력합니다.

```
# luupgrade -u -n BE_name -s os_image_path -j profile_path
```

-u 부트 환경에서 운영 체제 이미지를 업그레이드합니다.

-n BE_name 업그레이드될 부트 환경의 이름을 지정합니다.

- s *os_image_path* 운영 체제 이미지를 포함하는 디렉토리의 경로 이름을 지정합니다. 이 디렉토리는 DVD-ROM, CD-ROM과 같은 설치 매체에 있을 수도 있고 NFS 또는 UFS 디렉토리일 수도 있습니다.
- j *profile_path* 프로필에 대한 경로입니다. 프로필은 로컬 시스템의 디렉토리에 있어야 합니다. 프로필 만들기에 대한 자세한 내용은 96 페이지 “Solaris Live Upgrade에서 사용할 프로필 만들기”를 참조하십시오.

예 5-10 사용자 정의 JumpStart 프로필을 사용하여 부트 환경 업그레이드

이 예에서 `second_disk` 부트 환경은 프로필을 사용하여 업그레이드됩니다. `-j` 옵션은 프로필에 액세스하는 데 사용됩니다. 그러면 부트 환경이 활성화될 준비가 됩니다. 프로필을 만들려면 96 페이지 “Solaris Live Upgrade에서 사용할 프로필 만들기”를 참조하십시오. `pkgadd` 명령은 업그레이드하는 릴리스의 Solaris Live Upgrade 패키지를 추가합니다.

```
# pkgadd -d /server/packages SUNWlucfg SUNWlur SUNWluu
# luupgrade -u -n second_disk \
-s /net/installmachine/export/solarisX/OS_image \
-j /var/tmp/profile
```

부트 환경이 활성화될 준비가 되었습니다. 107 페이지 “부트 환경 활성화”를 참조하십시오.

부트 환경에 Solaris Flash 아카이브 설치

이 절에서는 Solaris Live Upgrade를 사용하여 Solaris Flash 아카이브를 설치하는 절차에 대해 설명합니다. Solaris Flash 아카이브를 설치하면 공유 파일을 제외하고 새 부트 환경에 있는 모든 파일을 덮어씁니다. 아카이브는 다음 매체에 저장됩니다.

- HTTP 서버
- FTP 서버 - 명령줄에서만 이 경로를 사용합니다.
- NFS 서버
- 로컬 파일
- 로컬 테이프
- DVD나 CD를 포함한 로컬 장치

Solaris Flash 아카이브를 설치하고 만드는 경우 다음 문제를 확인합니다.

<code>-n BE_name</code>	아카이브와 함께 설치될 부트 환경의 이름을 지정합니다.
<code>-s os_image_path</code>	운영 체제 이미지를 포함하는 디렉토리의 경로 이름을 지정합니다. 이 디렉토리는 DVD-ROM, CD-ROM과 같은 설치 매체에 있을 수도 있고 NFS 또는 UFS 디렉토리일 수도 있습니다. 이 OS 이미지는 Solaris Flash 아카이브를 쉽게 설치할 수 있도록 최소 부트 가능한 루트(/) 파일 시스템을 부트하는 미니루트를 제공합니다. 미니루트는 설치되는 이미지가 아닙니다. <code>-a</code> 옵션은 운영 체제 이미지를 제공합니다.
<code>-a archive</code>	아카이브가 로컬 파일 시스템에서 사용 가능한 경우 Solaris Flash 아카이브에 대한 경로입니다. <code>-s</code> 옵션과 <code>-a</code> 옵션을 사용하여 지정된 운영 시스템 이미지 버전은 반드시 동일해야 합니다.

예 5-11 부트 환경에 Solaris Flash 아카이브 설치

이 예에서 아카이브는 `second_disk` 부트 환경에 설치됩니다. 아카이브는 로컬 시스템에 있습니다. `-s` 옵션은 Solaris Flash 아카이브를 쉽게 설치할 수 있도록 최소 부트 가능한 루트(/) 파일 시스템을 부트하는 미니루트를 제공합니다. 미니루트는 설치되는 이미지가 아닙니다. `-a` 옵션은 운영 체제 이미지를 제공합니다. `-s` 및 `-a` 옵션용 운영 체제 버전은 모두 Solaris 10 5/09 릴리스입니다. 공유 가능 파일을 제외한 `second_disk`의 모든 파일을 덮어씁니다. `pkgadd` 명령은 업그레이드하는 릴리스의 Solaris Live Upgrade 패키지를 추가합니다.

```
# pkgadd -d /server/packages SUNWlucfg SUNWlur SUNWluu
# luupgrade -f -n second_disk \
-s /net/installmachine/export/Solaris_10/OS_image \
-a /net/server/archive/10
```

부트 환경이 활성화될 준비가 되었습니다. 107 페이지 “부트 환경 활성화”를 참조하십시오.

▼ 프로필을 사용하여 Solaris Flash 아카이브 설치

이 절차는 프로필을 사용하여 Solaris Flash 아카이브 또는 차등 아카이브를 설치하는 단계를 제공합니다.

프로필에 로케를 추가한 경우 추가 디스크 공간으로 부트 환경을 만들었는지 확인합니다.

- 1 **Solaris Live Upgrade** `SUNWlucfg`, `SUNWlur` 및 `SUNWluu` 패키지를 시스템에 설치합니다. 이 패키지는 업그레이드하는 릴리스에서 가져온 것이어야 합니다. 단계별 절차는 58 페이지 “`pkgadd` 명령을 사용하여 Solaris Live Upgrade 설치”를 참조하십시오.

2 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

3 프로필을 만듭니다.

Solaris Live Upgrade 프로필에서 사용할 수 있는 키워드 목록은 96 페이지 “Solaris Live Upgrade에서 사용할 프로필 만들기”를 참조하십시오.

4 다음을 입력합니다.

```
# luupgrade -f -n BE_name -s os_image_path -j profile_path
```

-f Solaris Flash 아카이브에서 운영 체제를 설치할 것을 나타냅니다.

-n *BE_name* 업그레이드될 부트 환경의 이름을 지정합니다.

-s *os_image_path* 운영 체제 이미지를 포함하는 디렉토리의 경로 이름을 지정합니다. 이 디렉토리는 DVD-ROM, CD-ROM과 같은 설치 매체에 있을 수도 있고 NFS 또는 UFS 디렉토리일 수도 있습니다. 이 OS 이미지는 Solaris Flash 아카이브를 쉽게 설치할 수 있도록 최소 부트 가능한 루트(/) 파일 시스템을 부트하는 미니루트를 제공합니다. 미니루트는 설치되는 이미지가 아닙니다. -j 옵션은 Solaris Flash 아카이브 운영 체제 이미지가 포함된 프로필의 경로를 제공합니다.

-j *profile_path* 플래시 설치를 위해 구성되는 JumpStart 프로필의 경로입니다. 프로필은 로컬 시스템의 디렉토리에 있어야 합니다. -s 옵션의 운영 체제 버전과 Solaris Flash 아카이브 운영 체제 버전이 같아야 합니다.

부트 환경이 활성화될 준비가 되었습니다. 107 페이지 “부트 환경 활성화”를 참조하십시오.

예 5-12 프로필을 사용하여 부트 환경에 Solaris Flash 아카이브 설치

이 예에서 프로필은 설치할 아카이브의 위치를 제공합니다.

```
# profile keywords      profile values
# -----
install_type           flash_install
archive_location      nfs installserver:/export/solaris/flasharchive/solarisarchive
```

프로필을 만든 후 `luupgrade` 명령을 실행하고 아카이브를 설치할 수 있습니다. -s 옵션은 Solaris Flash 아카이브를 쉽게 설치할 수 있도록 최소 부트 가능한 루트(/) 파일 시스템을 부트하는 미니루트를 제공합니다. 미니루트는 설치되는 이미지가 아닙니다. -j 옵션은

Solaris Flash 아카이브 운영 체제 이미지의 경로가 포함된 프로파일의 경로를 제공합니다. `-j` 옵션은 프로파일에 액세스하는 데 사용됩니다. `pkgadd` 명령은 업그레이드하는 릴리스의 Solaris Live Upgrade 패키지를 추가합니다.

```
# pkgadd -d /server/packages SUNWlucfg SUNWlur SUNWluu
# luupgrade -f -n second_disk \
-s /net/installmachine/export/solarisX/OS_image \
-j /var/tmp/profile
```

그러면 부트 환경이 활성화될 준비가 됩니다. 107 페이지 “부트 환경 활성화”를 참조하십시오.

프로파일을 만들려면 96 페이지 “Solaris Live Upgrade에서 사용할 프로파일 만들기”를 참조하십시오.

▼ 프로파일 키워드를 사용하여 Solaris Flash 아카이브 설치

이 절차를 수행하면 Solaris Flash 아카이브를 설치하고 프로파일 파일이 아닌 명령줄에서 `archive_location` 키워드를 사용합니다. 프로파일 파일을 사용하지 않고 신속하게 아카이브를 검색할 수 있습니다.

- 1 **Solaris Live Upgrade** `SUNWlucfg`, `SUNWlur` 및 `SUNWluu` 패키지를 시스템에 설치합니다. 이 패키지는 업그레이드하는 릴리스에서 가져온 것이어야 합니다. 단계별 절차는 58 페이지 “`pkgadd` 명령을 사용하여 Solaris Live Upgrade 설치”를 참조하십시오.
- 2 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 3 다음을 입력합니다.

```
# luupgrade -f -n BE_name -s os_image_path -J 'archive_location path-to-profile'
```

`-f` Solaris Flash 아카이브에서 운영 체제를 업그레이드하도록 지정합니다.

`-n BE_name` 업그레이드될 부트 환경의 이름을 지정합니다.

`-s os_image_path` 운영 체제 이미지를 포함하는 디렉토리의 경로 이름을 지정합니다. 이 디렉토리는 DVD-ROM, CD-ROM과 같은 설치 매체에 있을 수도 있고 NFS 또는 UFS 디렉토리일 수도 있습니다. 이 OS 이미지는 Solaris Flash 아카이브를 쉽게 설치할 수

있도록 최소 부트 가능한 루트(/) 파일 시스템을 부트하는 미니루트를 제공합니다. 미니루트는 설치되는 이미지가 아닙니다. -j 옵션은 Solaris Flash 아카이브 운영 체제 이미지가 포함된 프로필의 경로를 제공합니다.

-J 'archive_location path-to-profile'

archive_location 프로필 키워드와 JumpStart 프로필 경로를 지정합니다. -s 옵션의 운영 체제 버전과 Solaris Flash 아카이브 운영 체제 버전이 같아야 합니다. 키워드 값에 대한 자세한 내용은 **Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치**의 “archive_location 키워드”를 참조하십시오.

부트 환경이 활성화될 준비가 되었습니다. 107 페이지 “부트 환경 활성화”를 참조하십시오.

예 5-13 프로필 키워드를 사용하여 Solaris Flash 아카이브 설치

이 예에서 아카이브는 second_disk 부트 환경에 설치됩니다. -s 옵션은 Solaris Flash 아카이브를 쉽게 설치할 수 있도록 최소 부트 가능한 루트(/) 파일 시스템을 부트하는 미니루트를 제공합니다. 미니루트는 설치되는 이미지가 아닙니다. -j 옵션은 Solaris Flash 아카이브 운영 체제 이미지의 경로를 제공합니다. -J 옵션과 archive_location 키워드는 아카이브를 검색하는 데 사용합니다. 공유 가능 파일을 제외한 second_disk의 모든 파일을 덮어씁니다. pkgadd 명령은 업그레이드하는 릴리스의 Solaris Live Upgrade 패키지를 추가합니다.

```
# pkgadd -d /server/packages SUNWlucfg SUNWlur SUNWluu
# luupgrade -f -n second_disk \
-s /net/installmachine/export/solarisX/OS_image \
-J 'archive_location http://example.com/myflash.flar'
```

부트 환경 활성화

부트 환경을 활성화하면 다음에 시스템을 재부트할 때 해당 환경이 부트 가능하게 됩니다. 새 활성화 부트 환경을 부트할 때 오류가 발생하면 원래 부트 환경으로 신속하게 다시 전환할 수도 있습니다. 6 장, “오류 복구: 원래 부트 환경으로 폴백(작업)”을 참조하십시오.

설명	자세한 정보
<p>이 절차에 따라 <code>luactivate</code> 명령을 사용하여 부트 환경을 활성화합니다.</p> <p>주- 부트 환경을 처음 활성화할 때는 <code>luactivate</code> 명령을 사용해야 합니다.</p>	109 페이지 “부트 환경 활성화”
<p>이 절차에 따라 부트 환경을 활성화하고 파일을 강제로 동기화합니다.</p> <p>주- 파일이 첫 번째 활성화로 동기화됩니다. 처음 활성화된 후 부트 환경을 전환하면 파일이 동기화되지 않습니다.</p>	110 페이지 “부트 환경 활성화 및 파일 동기화”
<p>x86: 이 절차에 따라 GRUB 메뉴를 사용하여 부트 환경을 활성화합니다.</p> <p>주- GRUB 메뉴를 사용하면 한 부트 환경을 다른 부트 환경으로 쉽게 전환할 수 있습니다. 부트 환경은 처음 활성화된 후 GRUB 메뉴에 표시됩니다.</p>	112 페이지 “x86: GRUB 메뉴를 사용하여 부트 환경 활성화”

부트 환경 활성화에 대한 요구 사항 및 제한

부트 환경을 성공적으로 활성화하기 위해 해당 부트 환경은 다음 조건을 충족해야 합니다.

설명	자세한 정보
부트 환경은 "완료" 상태여야 합니다.	상태를 확인하려면 126 페이지 “모든 부트 환경의 상태 표시”를 참조하십시오.
부트 환경이 현재 부트 환경이 아닌 경우 <code>lumount</code> 또는 <code>mount</code> 명령을 사용하여 해당 부트 환경의 분할 영역을 마운트할 수 없습니다.	매뉴얼 페이지를 보려면 <code>lumount(1M)</code> 또는 <code>mount(1M)</code> 를 참조하십시오.
활성화하려는 부트 환경은 비교 작업에 포함될 수 없습니다.	절차에 대해서는 129 페이지 “부트 환경 비교”를 참조하십시오.
스왑을 재구성하려면 비활성 부트 환경을 부트하기 전에 이와 같이 변경합니다. 기본적으로 모든 부트 환경은 동일한 스왑 장치를 공유합니다.	스왑을 재구성하려면 68 페이지 “부트 환경 만들기 및 스왑 재구성”을 참조하십시오.

x86 전용 - x86 기반 시스템에서는 GRUB 메뉴를 사용하여 활성화할 수도 있습니다. 다음과 같은 예외가 있습니다.

- 부트 환경이 **Solaris 8, 9 또는 10 3/05 릴리스**로 만들어진 경우 부트 환경은 항상 `luactivate` 명령으로 활성화해야 합니다. 이러한 이전 부트 환경은 GRUB 메뉴에 표시되지 않습니다.
- 부트 환경을 처음 활성화할 때는 `luactivate` 명령을 사용해야 합니다. 다음에 부트하면 GRUB 기본 메뉴에 해당 부트 환경의 이름이 표시됩니다. 그런 후에 GRUB 메뉴에서 해당 항목을 선택하여 이 부트 환경으로 전환할 수 있습니다.

111 페이지 “x86: GRUB 메뉴를 사용하여 부트 환경 활성화”를 참조하십시오.

▼ 부트 환경 활성화

다음 절차에서는 새 부트 환경을 현재 실행 중인 부트 환경이 되도록 전환합니다.

x86 전용 - x86 기반 시스템에서는 GRUB 메뉴를 사용하여 활성화할 수도 있습니다. 다음과 같은 예외가 있습니다.

- 부트 환경이 **Solaris 8, 9 또는 10 3/05 릴리스**로 만들어진 경우 부트 환경은 항상 `luactivate` 명령으로 활성화해야 합니다. 이러한 이전 부트 환경은 GRUB 메뉴에 표시되지 않습니다.
- 부트 환경을 처음 활성화할 때는 `luactivate` 명령을 사용해야 합니다. 다음에 부트하면 GRUB 기본 메뉴에 해당 부트 환경의 이름이 표시됩니다. 그런 후에 GRUB 메뉴에서 해당 항목을 선택하여 이 부트 환경으로 전환할 수 있습니다.

111 페이지 “x86: GRUB 메뉴를 사용하여 부트 환경 활성화”를 참조하십시오.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 부트 환경을 활성화하려면 다음을 입력합니다.

```
# /sbin/luactivate BE_name
```

`BE_name` 활성화할 부트 환경의 이름을 지정합니다.

3 다시 부트합니다.

```
# init 6
```



주의 -init 또는 shutdown 명령만 사용하여 재부트합니다. reboot, halt 또는 uadmin 명령을 사용하면 시스템이 부트 환경을 전환하지 않습니다. 마지막 활성화 부트 환경이 다시 부트됩니다.

예 5-14 부트 환경 활성화

이 예에서 다음 재부트할 때 second_disk 부트 환경이 활성화됩니다.

```
# /sbin/luactivate second_disk
# init 6
```

▼ 부트 환경 활성화 및 파일 동기화

새로 만든 부트 환경에서 처음 부트할 때 Solaris Live Upgrade 소프트웨어가 새 부트 환경을 마지막 활성화 부트 환경과 동기화합니다. "동기화"란 특정 핵심 시스템 파일과 디렉토리가 가장 최신의 활성화 부트 환경에서 부트 중인 부트 환경으로 복사됨을 의미합니다. Solaris Live Upgrade는 luactivate 명령과 -s 옵션을 사용하여 동기화를 강제할 경우가 아니면 처음 부트한 후에 이와 같은 동기화를 수행하지 않습니다.

x86 전용 - GRUB 메뉴를 사용하여 부트 환경을 전환하는 경우에도 파일이 동기화되지 않습니다. 파일을 동기화하려면 다음 절차를 사용해야 합니다.

동기화에 대한 자세한 내용은 52 페이지 “부트 환경 사이의 파일 동기화”를 참조하십시오.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 부트 환경을 활성화하려면 다음을 입력합니다.

```
# /sbin/luactivate -s BE_name
```

-s 마지막 활성화 부트 환경과 새 부트 환경 사이에서 파일의 동기화를 강제합니다. 처음으로 부트 환경이 활성화되면 부트 환경 간의 파일이 동기화되지만 이후에는 -s 옵션을 사용한 경우가 아니면 파일이 동기화되지 않습니다.



주의 - 마지막 활성화 부트 환경에서 발생했을 수 있는 변경 사항을 인식하거나 제어하지 못할 수 있으므로 이 옵션을 사용할 때에는 신중해야 합니다. 예를 들어, 현재 부트 환경에서 Solaris 10 5/09 소프트웨어를 실행하고 있으며 강제 동기화하여 Solaris 9 릴리스로 다시 부트하는 경우 파일이 Solaris 9 릴리스에서 변경될 수 있습니다. 파일은 OS의 릴리스에 따라 달라지므로 Solaris 9 릴리스로의 부트가 실패할 수 있는데, 이는 Solaris 10 5/09 파일이 Solaris 9 파일과 호환되지 않을 수 있기 때문입니다.

`BE_name` 활성화할 부트 환경의 이름을 지정합니다.

3 다시 부트합니다.

```
# init 6
```

예 5-15 부트 환경 활성화

이 예에서 다음 재부트할 때 `second_disk` 부트 환경이 활성화되며 파일이 동기화됩니다.

```
# /sbin/luactivate -s second_disk
# init 6
```

x86: GRUB 메뉴를 사용하여 부트 환경 활성화

GRUB 메뉴는 부트 환경을 전환할 수 있는 선택적 방법을 제공합니다. GRUB 메뉴는 `luactivate` 명령을 사용한 활성화(부트) 방법의 대안입니다. 아래 표에서는 GRUB 메뉴를 사용할 때의 주의 사항과 제한 사항을 보여 줍니다.

표 5-3 x86: GRUB 메뉴를 사용하여 활성화 요약

작업	설명	자세한 정보
주의	부트 환경을 활성화한 후에는 BIOS에서 디스크 순서를 변경하지 마십시오. 순서를 변경하면 GRUB 메뉴가 무효화될 수 있습니다. 이 문제가 발생하면 디스크 순서를 다시 원래 상태로 변경하여 GRUB 메뉴를 수정합니다.	
처음으로 부트 환경 활성화	부트 환경을 처음 활성화할 때는 <code>luactivate</code> 명령을 사용해야 합니다. 다음에 부트하면 GRUB 기본 메뉴에 해당 부트 환경의 이름이 표시됩니다. 그런 후에 GRUB 메뉴에서 해당 항목을 선택하여 이 부트 환경으로 전환할 수 있습니다.	109 페이지 “부트 환경 활성화”

표 5-3 x86: GRUB 메뉴를 사용하여 활성화 요약 (계속)

작업	설명	자세한 정보
파일 동기화	처음으로 부트 환경을 활성화하면 현재 부트 환경과 새 부트 환경 간에 파일이 동기화됩니다. 이후 활성화 시에는 파일이 동기화되지 않습니다. GRUB 메뉴를 사용하여 부트 환경을 전환하는 경우에도 파일이 동기화되지 않습니다. <code>luactivate</code> 명령을 <code>-s</code> 옵션과 함께 사용하면 강제로 동기화할 수 있습니다.	110 페이지 “부트 환경 활성화 및 파일 동기화”
Solaris 10 1/06 릴리스 전에 생성된 부트 환경	부트 환경이 Solaris 8, 9 또는 10 3/05 릴리스로 만들어진 경우 부트 환경은 항상 <code>luactivate</code> 명령으로 활성화해야 합니다. 이러한 이전 부트 환경은 GRUB 메뉴에 표시되지 않습니다.	109 페이지 “부트 환경 활성화”
GRUB 메뉴 항목 편집 또는 사용자 정의	<p><code>menu.lst</code> 파일에는 GRUB 메뉴에 표시되는 정보가 들어 있습니다. 다음과 같은 이유로 이 파일을 변경할 수 있습니다.</p> <ul style="list-style-type: none"> ■ Solaris OS가 아닌 운영 체제에 대한 항목을 GRUB 메뉴에 추가 ■ 부트 동작을 사용자 정의. 예를 들어, 자동으로 OS를 부트하는 기본 시간을 변경하거나 부트를 자세한 표시 모드로 변경할 수 있습니다. <p>주 - GRUB 메뉴를 변경하려면 <code>menu.lst</code> 파일을 찾아야 합니다. 단계별 지침은 System Administration Guide: Basic Administration의 14 장, “Managing the Solaris Boot Archives (Tasks)”를 참조하십시오.</p> <p>주의 - GRUB <code>menu.lst</code> 파일을 사용하여 Solaris Live Upgrade 항목을 수정하지 마십시오. 이렇게 하면 Solaris Live Upgrade가 실패할 수 있습니다. <code>menu.lst</code> 파일을 사용하여 부트 동작을 사용자 정의할 수도 있지만 기본 사용자 정의 방법은 <code>eeprom</code> 명령을 사용하는 것입니다. <code>menu.lst</code> 파일을 사용하여 사용자 정의하는 경우 소프트웨어 업그레이드 중에 Solaris OS 항목을 수정할 수 있습니다. 이 경우 파일 변경 사항이 손실될 수 있습니다.</p>	<ul style="list-style-type: none"> ■ Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획의 “GRUB 기반 부트(개요)” ■ System Administration Guide: Basic Administration의 “Booting an x86 Based System by Using GRUB (Task Map)”

▼ x86: GRUB 메뉴를 사용하여 부트 환경 활성화

GRUB 메뉴를 사용하여 부트 환경 간에 전환할 수 있습니다. 다음 제한 사항을 확인합니다.

- 첫 번째 부트 환경의 활성화는 `luactivate` 명령으로 수행해야 합니다. 초기 활성화 후 GRUB 메뉴에 부트 환경이 표시됩니다. 그런 다음에는 GRUB 메뉴에서 부트 환경을 부트할 수 있습니다.

- 주의 - GRUB 메뉴를 사용하여 부트 환경으로 전환하면 동기화가 수행되지 않습니다. 파일 동기화에 대한 자세한 내용은 53 페이지 “부트 환경 간의 강제 동기화” 링크를 참조하십시오.
- 부트 환경이 Solaris 8, 9 또는 10 3/05 릴리스로 만들어진 경우 부트 환경은 항상 luactivate 명령으로 활성화해야 합니다. 이러한 이전 부트 환경은 GRUB 메뉴에 표시되지 않습니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 시스템을 다시 부트합니다.

```
# init 6
```

GRUB 기본 메뉴가 표시됩니다. Solaris Live Upgrade 부트 환경인 second_disk와 Solaris 등 두 개의 운영 체제가 나열됩니다. 어떤 이유로 주 OS가 부트되지 않을 경우 failsafe 항목이 복구에 사용됩니다.

```
GNU GRUB version 0.95 (616K lower / 4127168K upper memory)
```

```
+-----+
|Solaris                               |
|Solaris failsafe                      |
|second_disk                           |
|second_disk failsafe                  |
+-----+
```

```
Use the ^ and v keys to select which entry is highlighted. Press
enter to boot the selected OS, 'e' to edit the commands before
booting, or 'c' for a command-line.
```

3 부트 환경을 활성화하려면 화살표 키를 사용하여 원하는 부트 환경을 선택한 다음 Enter를 누릅니다.

선택한 부트 환경이 부트되어 활성화 부트 환경이 됩니다.

오류 복구: 원래 부트 환경으로 폴백(작업)

이 장에서는 활성화 오류에서 복구하는 방법에 대해 설명합니다.

주 - 이 장에서는 UFS 파일 시스템용 Solaris Live Upgrade에 대해 설명합니다. 사용법은 ZFS 부트 환경에서의 `luactivate` 명령과 동일합니다. UFS 파일 시스템을 ZFS 루트 폴로 마이그레이션하거나 ZFS 루트 폴을 만들고 설치하는 절차에 대한 자세한 내용은 13 장, “ZFS 루트 폴의 부트 환경 만들기”를 참조하십시오.

업그레이드 후 실패가 발견되거나 업그레이드된 구성 요소와 응용 프로그램이 호환되지 않는 경우 플랫폼에 따라 다음 절차 중 하나를 사용하여 원래 부트 환경으로 폴백할 수 있습니다.

- SPARC 기반 시스템의 경우:
 - 116 페이지 “SPARC: 성공적인 새 부트 환경 활성화 시의 폴백”
 - 116 페이지 “SPARC: 실패한 부트 환경 활성화에서의 폴백”
 - 117 페이지 “SPARC: DVD, CD 또는 네트워크 설치 이미지를 사용하여 원래 부트 환경으로 폴백”
- x86 기반 시스템의 경우:
 - 118 페이지 “x86: GRUB 메뉴를 사용하여 성공적인 새 부트 환경 활성화 시 폴백”
 - 120 페이지 “x86: GRUB 메뉴를 사용하여 실패한 부트 환경 활성화 시의 폴백”
 - 122 페이지 “x86: GRUB 메뉴와 DVD 또는 CD를 사용하여 실패한 부트 환경 활성화에서의 폴백”

SPARC: 원래 부트 환경으로 폴백

다음 세 가지 방법을 사용하여 원래 부트 환경으로 폴백할 수 있습니다.

- 116 페이지 “SPARC: 성공적인 새 부트 환경 활성화 시의 폴백”
- 116 페이지 “SPARC: 실패한 부트 환경 활성화에서의 폴백”
- 117 페이지 “SPARC: DVD, CD 또는 네트워크 설치 이미지를 사용하여 원래 부트 환경으로 폴백”

▼ SPARC: 성공적인 새 부트 환경 활성화 시의 폴백

새 부트 환경을 성공적으로 활성화했지만 결과에 만족하지 않을 경우 이 절차를 사용합니다.

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# /sbin/luactivate BE_name
```

BE_name 활성화할 부트 환경의 이름을 지정합니다.

- 3 다시 부트합니다.

```
# init 6
```

이전에 작동한 부트 환경이 활성 부트 환경이 됩니다.

▼ SPARC: 실패한 부트 환경 활성화에서의 폴백

- 새 부트 환경을 부트 중에 오류가 발생하고 단일 사용자 모드에서 원래 부트 환경을 부트할 수 있을 경우 이 절차를 사용하여 원래 부트 환경으로 폴백합니다.
- 매체 또는 네트워크 설치 이미지에서 부트해야 하는 경우 [117 페이지 “SPARC: DVD, CD 또는 네트워크 설치 이미지를 사용하여 원래 부트 환경으로 폴백”](#)을 참조하십시오.

- 1 OK 프롬프트에서 **Solaris Operating System DVD, Solaris Software - 1 CD**, 네트워크 또는 로컬 디스크에서 단일 사용자 상태로 시스템을 부트합니다.

```
OK boot device_name -s
```

device_name 시스템이 부트할 수 있는 장치의 이름, 예를 들어 /dev/dsk/c0t0d0s0을 지정합니다.

2 다음을 입력합니다.

```
# /sbin/luactivate BE_name
```

BE_name 활성화할 부트 환경의 이름을 지정합니다.

- 이 명령으로 프롬프트를 표시할 수 없는 경우 117 페이지 “SPARC: DVD, CD 또는 네트워크 설치 이미지를 사용하여 원래 부트 환경으로 폴백”을 계속합니다.
- 프롬프트가 표시되면 계속합니다.

3 프롬프트에서 다음을 입력합니다.

```
Do you want to fallback to activate boot environment <disk name>
(yes or no)? yes
```

폴백 활성화가 성공했다는 메시지가 표시합니다.

4 다시 부트합니다.

```
# init 6
```

이전에 작동한 부트 환경이 활성화 부트 환경이 됩니다.

▼ SPARC: DVD, CD 또는 네트워크 설치 이미지를 사용하여 원래 부트 환경으로 폴백

이 절차를 사용하여 DVD, CD, 네트워크 설치 이미지 또는 부트할 수 있는 다른 디스크에서 부트할 수 있습니다. 마지막 활성화 부트 환경에서 `root()` 슬라임을 마운트해야 합니다. 그런 다음 전환을 수행하는 `luactivate` 명령을 실행합니다. 다시 부트하면 마지막 활성화 부트 환경이 다시 실행됩니다.

1 OK 프롬프트에서 Solaris Operating System DVD, Solaris Software - 1 CD, 네트워크 또는 로컬 디스크에서 단일 사용자 상태로 시스템을 부트합니다.

```
OK boot cdrom -s
```

또는

```
OK boot net -s
```

또는

```
OK boot device_name -s
```

device_name 운영 체제의 복사본이 상주하는 디스크 및 슬라이스의 이름, 예를 들어 `/dev/dsk/c0t0d0s0`을 지정합니다.

2 필요할 경우 폴백 부트 환경에 대한 루트(/) 파일 시스템의 무결성을 검사합니다.

```
# fsck device_name
```

device_name 폴백하려는 부트 환경의 디스크 장치에 있는 루트(/) 파일 시스템의 위치를 지정합니다. 장치 이름은 /dev/dsk/c wtxd ysz의 형식으로 입력됩니다.

3 **활성 부트 환경 루트(/) 슬라이스를 디렉토리(예:/mnt)로 마운트합니다.**

```
# mount device_name /mnt
```

device_name 폴백하려는 부트 환경의 디스크 장치에 있는 루트(/) 파일 시스템의 위치를 지정합니다. 장치 이름은 /dev/dsk/c wtxd ysz의 형식으로 입력됩니다.

4 **활성 부트 환경 루트(/) 슬라이스에서 다음을 입력합니다.**

```
# /mnt/sbin/luactivate
```

luactivate가 이전 작동 부트 환경을 활성화하고 결과를 나타냅니다.

5 **/mnt를 마운트 해제합니다.**

```
# umount /mnt
```

6 **다시 부트합니다.**

```
# init 6
```

이전에 작동한 부트 환경이 활성 부트 환경이 됩니다.

x86: 원래 부트 환경으로 폴백

원래 부트 환경으로 폴백하려면 사용자 환경에 가장 적합한 절차를 선택합니다.

- 118 페이지 “x86: GRUB 메뉴를 사용하여 성공적인 새 부트 환경 활성화 시 폴백”
- 120 페이지 “x86: GRUB 메뉴를 사용하여 실패한 부트 환경 활성화 시의 폴백”
- 122 페이지 “x86: GRUB 메뉴와 DVD 또는 CD를 사용하여 실패한 부트 환경 활성화에서의 폴백”

▼ x86: GRUB 메뉴를 사용하여 성공적인 새 부트 환경 활성화 시 폴백

새 부트 환경을 성공적으로 활성화했지만 결과에 만족하지 않을 경우 이 절차를 사용합니다. GRUB 메뉴를 사용하여 원래 부트 환경으로 빨리 전환할 수 있습니다.

주 - 전환되는 부트 환경은 GRUB 소프트웨어를 사용하여 만든 GRUB 부트 환경이어야 합니다. **Solaris 8, 9 또는 10 3/05 릴리스**에서 부트 환경을 만든 경우 부트 환경은 GRUB 부트 환경이 아닙니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 시스템을 다시 부트합니다.

```
# init 6
```

GRUB 메뉴가 표시됩니다. 원래 부트 환경은 Solaris OS입니다. `second_disk` 부트 환경이 성공적으로 활성화되었으며 GRUB 메뉴에 표시됩니다. 어떤 이유로 주 항목이 부트되지 않을 경우 `failsafe` 항목이 복구에 사용됩니다.

```
GNU GRUB version 0.95 (616K lower / 4127168K upper memory)
+-----+
|Solaris                                     |
|Solaris failsafe                           |
|second_disk                                 |
|second_disk failsafe                       |
+-----+
Use the ^ and v keys to select which entry is highlighted. Press
enter to boot the selected OS, 'e' to edit the commands before
booting, or 'c' for a command-line.
```

3 원래 부트 환경으로 부트하려면 화살표 키를 사용하여 원래 부트 환경을 선택한 다음 Enter를 누릅니다.

예 6-1 성공적인 새 부트 환경 활성화 시의 폴백

```
# su
# init 6

GNU GRUB version 0.95 (616K lower / 4127168K upper memory)
+-----+
|Solaris                                     |
|Solaris failsafe                           |
|second_disk                                 |
|second_disk failsafe                       |
+-----+
Use the ^ and v keys to select which entry is highlighted. Press
```

enter to boot the selected OS, 'e' to edit the commands before booting, or 'c' for a command-line.

원래 부트 환경인 Solaris를 선택합니다.

▼ x86: GRUB 메뉴를 사용하여 실패한 부트 환경 활성화 시의 폴백

부트 중에 오류가 발생하면 다음 절차에 따라 원래 부트 환경으로 폴백합니다. 이 예에서 GRUB 메뉴는 올바르게 표시되지만 새 부트 환경이 부트 가능하지 않습니다. 장치는 /dev/dsk/c0t4d0s0입니다. 원래 부트 환경인 c0t4d0s0이 활성화 부트 환경이 됩니다.



주의 - Solaris 10 3/05 릴리스의 경우 이전 부트 환경과 새 부트 환경이 다른 디스크에 있을 때 권장되는 폴백 방법으로 BIOS에서 하드 디스크 부트 순서를 변경하는 방법도 있습니다. Solaris 10 1/06 릴리스부터 BIOS 디스크 순서를 변경할 필요가 없으며 변경하지 않는 것이 좋습니다. BIOS 디스크 순서를 변경하면 GRUB 메뉴가 무효화되며 부트 환경이 부트 불가능하게 됩니다. BIOS 디스크 순서가 변경된 경우 순서를 다시 원래 설정으로 되돌리면 시스템 기능이 복원됩니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 GRUB 메뉴를 표시하려면 시스템을 재부트합니다.

```
# init 6
```

GRUB 메뉴가 표시됩니다.

```
GNU GRUB version 0.95 (616K lower / 4127168K upper memory)
+-----+
|Solaris                                     |
|Solaris failsafe                           |
|second_disk                                 |
|second_disk failsafe                        |
+-----+
Use the ^ and v keys to select which entry is highlighted. Press
enter to boot the selected OS, 'e' to edit the commands before
booting, or 'c' for a command-line.
```

3 GRUB 메뉴에서 원래 부트 환경을 선택합니다. GRUB 소프트웨어를 사용하여 이미 부트 환경이 만들어져 있어야 합니다. Solaris 10 1/06 릴리스 이전에 작성된 부트 환경은 GRUB

부트 환경이 아닙니다. 부트 가능한 GRUB 부트 환경이 없으면 122 페이지 “x86: GRUB 메뉴와 DVD 또는 CD를 사용하여 실패한 부트 환경 활성화에서의 폴백” 절차로 건너뛰니다.

4 GRUB 메뉴를 편집하여 단일 사용자 모드로 부트합니다.

a. GRUB 기본 메뉴를 편집하려면 **e**를 입력합니다.

GRUB 편집 메뉴가 표시됩니다.

```
root (hd0,2,a)
kernel /platform/i86pc/multiboot
module /platform/i86pc/boot_archive
```

b. 화살표 키를 사용하여 원래 부트 환경의 커널 항목을 선택합니다.

c. 부트 항목을 편집하려면 **e**를 입력합니다.

GRUB 편집 메뉴에 커널 항목이 표시됩니다.

```
grub edit>kernel /boot/multiboot
```

d. **-s**를 입력한 다음 **Enter**를 누릅니다.

다음 예에서 **-s** 옵션의 위치를 확인합니다.

```
grub edit>kernel /boot/multiboot -s
```

e. 단일 사용자 모드로 부트 프로세스를 시작하려면 **b**를 입력합니다.

5 필요할 경우 폴백 부트 환경에 대한 루트(/) 파일 시스템의 무결성을 검사합니다.

```
# fsck mount_point
```

mount_point 알려져 있고 신뢰성 있는 루트(/) 파일 시스템입니다.

6 원래 부트 환경 루트 슬라이스를 디렉토리(예: /mnt)로 마운트합니다.

```
# mount device_name /mnt
```

device_name 폴백하려는 부트 환경의 디스크 장치에 있는 루트(/) 파일 시스템의 위치를 지정합니다. 장치 이름은 /dev/dsk/c wtxd ysz의 형식으로 입력됩니다.

7 활성화 부트 환경 루트 슬라이스에서 다음을 입력합니다.

```
# /mnt/sbin/luactivate
```

luactivate가 이전 작동 부트 환경을 활성화하고 결과를 나타냅니다.

8 /mnt를 마운트 해제합니다.

```
# umount /mnt
```

9 다시 부트합니다.

init 6

이전에 작동한 부트 환경이 활성화 부트 환경이 됩니다.

▼ x86: GRUB 메뉴와 DVD 또는 CD를 사용하여 실패한 부트 환경 활성화에서의 폴백

부트 중에 오류가 발생하면 다음 절차에 따라 원래 부트 환경으로 폴백합니다. 이 예에서는 새 부트 환경을 부트할 수 없는 상태입니다. 또한 GRUB 메뉴가 표시되지 않습니다. 장치는 /dev/dsk/c0t4d0s0입니다. 원래 부트 환경인 c0t4d0s0이 활성화 부트 환경이 됩니다.



주의 - Solaris 10 3/05 릴리스의 경우 이전 부트 환경과 새 부트 환경이 다른 디스크에 있을 때 권장되는 폴백 방법으로 BIOS에서 하드 디스크 부트 순서를 변경하는 방법도 있습니다. Solaris 10 1/06 릴리스부터 BIOS 디스크 순서를 변경할 필요가 없으며 변경하지 않는 것이 좋습니다. BIOS 디스크 순서를 변경하면 GRUB 메뉴가 무효화되며 부트 환경이 부트 불가능하게 됩니다. BIOS 디스크 순서가 변경된 경우 순서를 다시 원래 설정으로 되돌리면 시스템 기능이 복원됩니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 Solaris Operating System for x86 Platforms DVD 또는 Solaris Software for x86 Platforms - 1 CD를 넣습니다.

3 DVD 또는 CD에서 부트합니다.

init 6

GRUB 메뉴가 표시됩니다.

```
GNU GRUB version 0.95 (616K lower / 4127168K upper memory)
+-----+
|Solaris 10 5/09                               |
|Solaris 10 5/09 Serial Console ttya          |
|Solaris 10 5/09 Serial Console ttyb (for lx50, v60x and v65x |
+-----+
Use the ^ and v keys to select which entry is highlighted. Press
enter to boot the selected OS, 'e' to edit the commands before
booting, or 'c' for a command-line.
```

- 4 기본 옵션으로 부트할 때까지 기다리거나 표시된 옵션을 모두 선택합니다.
설치 화면이 표시됩니다.

```
+-----+
|Select the type of installation you want to perform:      |
|                                                           |
|      1 Solaris Interactive                               |
|      2 Custom JumpStart                                 |
|      3 Solaris Interactive Text (Desktop session)       |
|      4 Solaris Interactive Text (Console session)      |
|      5 Apply driver updates                             |
|      6 Single user shell                               |
|                                                           |
|      Enter the number of your choice followed by the <ENTER> key. |
|      Alternatively, enter custom boot arguments directly. |
|                                                           |
|      If you wait 30 seconds without typing anything,   |
|      an interactive installation will be started.      |
+-----+
```

- 5 “Single user shell” 옵션을 선택합니다.

다음 메시지가 표시됩니다.

```
Do you wish to automatically update the boot archive? y /n
```

- 6 다음을 입력합니다.n

```
Starting shell...
```

```
#
```

사용자는 이제 단일 사용자 모드에 있습니다.

- 7 필요할 경우 폴백 부트 환경에 대한 루트(/) 파일 시스템의 무결성을 검사합니다.

```
# fsck mount_point
```

mount_point 알려져 있고 신뢰성 있는 루트(/) 파일 시스템입니다.

- 8 원래 부트 환경 루트 슬라이스를 디렉토리(예: /mnt)로 마운트합니다.

```
# mount device_name /mnt
```

device_name 폴백하려는 부트 환경의 디스크 장치에 있는 루트(/) 파일 시스템의 위치를 지정합니다. 장치 이름은 /dev/dsk/c wtxd ysz의 형식으로 입력됩니다.

9 활성화 부트 환경 루트 슬라이스에서 다음을 입력합니다.

```
# /mnt/sbin/luactivate
Do you want to fallback to activate boot environment c0t4d0s0
(yes or no)? yes
```

luactivate가 이전 작동 부트 환경을 활성화하고 결과를 나타냅니다.

10 /mnt를 마운트 해제합니다.

```
# umount device_name
```

device_name 폴백하려는 부트 환경의 디스크 장치에 있는 루트(/) 파일 시스템의 위치를 지정합니다. 장치 이름은 /dev/dsk/c wtxd ysz의 형식으로 입력됩니다.

11 다시 부트합니다.

```
# init 6
```

이전에 작동한 부트 환경이 활성화 부트 환경이 됩니다.

Solaris Live Upgrade 부트 환경 유지 관리(작업)

이 장에서는 부트 환경 파일 시스템을 최신으로 유지하거나 부트 환경을 삭제하는 등의 다양한 관리 작업에 대해 설명합니다. 이 장은 다음 절로 구성되어 있습니다.

주 - 이 장에서는 UFS 파일 시스템용 Solaris Live Upgrade에 대해 설명합니다. ZFS 부트 환경에서의 유지 관리 사용법은 동일합니다. UFS 파일 시스템을 ZFS 루트 풀로 마이그레이션하거나 ZFS 루트 풀을 만들고 설치하는 절차에 대한 자세한 내용은 13 장, “ZFS 루트 풀의 부트 환경 만들기”를 참조하십시오.

- 125 페이지 “Solaris Live Upgrade 유지 관리 개요”
- 126 페이지 “모든 부트 환경의 상태 표시”
- 127 페이지 “사전 구성된 부트 환경 업데이트”
- 128 페이지 “예약된 만들기, 업그레이드 또는 복사 작업 취소”
- 129 페이지 “부트 환경 비교”
- 130 페이지 “비활성 부트 환경 삭제”
- 131 페이지 “활성 부트 환경의 이름 표시”
- 131 페이지 “부트 환경의 이름 변경”
- 133 페이지 “부트 환경 이름과 관련된 설명 추가 또는 변경”
- 136 페이지 “부트 환경의 구성 보기”

Solaris Live Upgrade 유지 관리 개요

표 7-1 Solaris Live Upgrade 유지 관리 개요

작업	설명	수행 방법
(선택 사항) 상태 보기.	<ul style="list-style-type: none"> ■ 부트 환경이 현재 사용 중인지, 활성화되는 중인지 또는 비교되는 중인지를 확인합니다. 	<ul style="list-style-type: none"> ■ 126 페이지 “모든 부트 환경의 상태 표시”

표 7-1 Solaris Live Upgrade 유지 관리 개요 (계속)

작업	설명	수행 방법
	<ul style="list-style-type: none"> ■ 활성 및 비활성 부트 환경을 비교합니다. ■ 활성 부트 환경의 이름을 표시합니다. ■ 부트 환경의 구성을 표시합니다. 	<ul style="list-style-type: none"> ■ 129 페이지 “부트 환경 비교” ■ 131 페이지 “활성 부트 환경의 이름 표시” ■ 136 페이지 “부트 환경의 구성 보기”
(선택 사항) 비활성 부트 환경 업데이트하기.	파일 시스템의 구성을 변경하지 않고 다시 활성 부트 환경에서 파일 시스템을 복사합니다.	127 페이지 “사전 구성된 부트 환경 업데이트”
(선택 사항) 기타 작업.	<ul style="list-style-type: none"> ■ 부트 환경을 삭제합니다. ■ 부트 환경의 이름을 변경합니다. ■ 부트 환경 이름과 연결된 설명을 추가 또는 변경합니다. ■ 예약된 작업을 취소합니다. 	<ul style="list-style-type: none"> ■ 130 페이지 “비활성 부트 환경 삭제” ■ 131 페이지 “부트 환경의 이름 변경” ■ 133 페이지 “부트 환경 이름과 관련된 설명 추가 또는 변경” ■ 128 페이지 “예약된 만들기, 업그레이드 또는 복사 작업 취소”

모든 부트 환경의 상태 표시

lustatus 명령을 사용하여 부트 환경에 대한 정보를 표시합니다. 어떤 부트 환경도 지정되지 않았다면 시스템이 모든 부트 환경에 관한 상태 정보가 표시됩니다.

각 부트 환경별로 다음과 같은 세부 사항이 표시됩니다.

- 이름 - 각 부트 환경의 이름입니다.
- 완료 - 진행 중인 복사 또는 만들기 작업이 없음을 나타냅니다. 또한 부트 환경이 부트될 수 있습니다. 만들기 또는 업그레이드 작업의 현재 활동 또는 오류는 불완전한 부트 환경을 유발할 수 있습니다. 예를 들어 부트 환경에 대하여 복사 작업이 진행 중이거나 예약되어 있는 경우 해당 부트 환경은 완전하지 않은 것으로 간주됩니다.
- 활성 - 활성 부트 환경인지 여부를 나타냅니다.
- ActiveOnReboot - 다음에 시스템을 재부트하면 부트 환경이 활성화되는지 여부를 나타냅니다.
- CopyStatus - 부트 환경의 만들기 또는 복사가 예정되어 있는지, 활성 상태인지 또는 업그레이드 중인지 여부를 나타냅니다. SCHEDULED 상태는 사용자가 Live Upgrade 복사, 이름 바꾸기 또는 업그레이드 작업을 수행하지 못하도록 합니다.

▼ 모든 부트 환경의 상태 표시

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# lustatus BE_name
```

BE_name 상태를 표시할 비활성 부트 환경의 이름을 지정합니다. *BE_name*을 생략하면 `lustatus`는 시스템의 모든 부트 환경을 위한 상태를 표시합니다.

이 예에서는 모든 부트 환경을 위한 상태가 표시됩니다.

```
# lustatus
```

boot environment Name	Is Complete	Active Now	Active OnReboot	Can Delete	Copy Status
disk_a_S9	yes	yes	yes	no	-
disk_b_S10database	yes	no	no	yes	COPYING
disk_b_S9a	no	no	no	yes	-

주-disk_b_S9a는 완료되지 않았고, disk_b_S10database는 Live Upgrade 작업이 진행 중이기 때문에 복사, 이름 바꾸기 또는 업그레이드 작업을 수행할 수 없습니다.

사전 구성된 부트 환경 업데이트

복사 메뉴 또는 `lumake` 명령을 사용하여 이전에 구성된 부트 환경의 내용을 업데이트할 수 있습니다. 활성(소스) 부트 환경의 파일 시스템이 대상 부트 환경으로 복사됩니다. 대상의 데이터도 완전히 삭제됩니다. 특정 부트 환경에서 복사하려면 먼저 해당 부트 환경이 "완료" 상태여야 합니다. 부트 환경의 상태를 확인하려면 [126 페이지 “모든 부트 환경의 상태 표시”](#)를 참조하십시오.

복사 작업을 나중에 수행하도록 예약할 수 있으며 한 번에 하나의 작업만 예약할 수 있습니다. 예약된 복사를 취소하려면 [128 페이지 “예약된 만들기, 업그레이드 또는 복사 작업 취소”](#)를 참조하십시오.

▼ 이전에 구성된 부트 환경 업데이트

이 절차에서는 소스 파일을 이전에 만든 부트 환경의 오래된 파일에 겹쳐서 복사합니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 다음을 입력합니다.

```
# lumake -n BE_name [-s source_BE] [-t time] [-m email_address]
```

-n *BE_name* 교체할 파일 시스템을 갖고 있는 부트 환경의 이름을 지정합니다.

-s *source_BE* (선택 사항) 대상 부트 환경으로 복사할 파일 시스템을 포함하고 있는 소스 부트 환경의 이름을 지정합니다. 이 옵션을 생략하면 lumake는 현재 부트 환경을 소스로 사용합니다.

-t *time* (선택 사항) 지정된 부트 환경의 파일 시스템에 지정된 시간에 복사하도록 일괄 작업을 설정합니다. 시간은 at (1) 매뉴얼 페이지에 지정되어 있는 형식으로 제공됩니다.

-m *email_address* (선택 사항) 명령이 완료될 때 lumake 출력을 지정된 주소로 전자 메일로 보낼 수 있습니다. *email_address*는 검사하지 않습니다. 이 옵션은 -t와 함께만 사용할 수 있습니다.

예 7-1 사전 구성된 부트 환경 업데이트

이 예에서 *first_disk*의 파일 시스템은 *second_disk*로 복사됩니다. 작업이 완료되면 anywhere.com의 Joe에게 전자 메일이 전송됩니다.

```
# lumake -n second_disk -s first_disk -m joe@anywhere.com
```

*first_disk*의 파일들은 *second_disk*로 복사되고 알림을 위해 전자 우편이 전송됩니다. 예약된 복사를 취소하려면 128 페이지 “예약된 만들기, 업그레이드 또는 복사 작업 취소”를 참조하십시오.

예약된 만들기, 업그레이드 또는 복사 작업 취소

부트 환경의 예약된 만들기, 업그레이드 또는 복사 작업은 작업 시작 직전에 취소할 수 있습니다. lumake 명령으로 작업을 예약할 수 있습니다. 시스템에서 항상 하나의 작업만 예약할 수 있습니다.

▼ 예약된 만들기, 업그레이드 또는 복사 작업 취소

- 1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# lucancel
```

해당 작업은 이제 지정된 시간에 더 이상 수행되지 않습니다.

부트 환경 비교

lucompare 명령을 사용하여 활성 부트 환경과 다른 부트 환경 간의 차이를 확인합니다. 비교하려면 비활성 부트 환경이 완료 상태여야 하며 대기 중인 복사 작업이 있으면 안 됩니다. [126 페이지](#) “모든 부트 환경의 상태 표시”를 참조하십시오.

lucompare 명령은 모든 비전역 영역의 내용이 포함된 부트 환경의 비교를 생성합니다.

지정된 부트 환경은 lumount 또는 mount를 사용하여 마운트된 어떤 분할 영역도 갖고 있으면 안 됩니다.

▼ 부트 환경 비교

- 1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# /usr/sbin/lucompare -i infile (or) -t -o outfile BE_name
```

-i infile *infile*에 나열된 파일들을 비교합니다. 비교할 파일의 이름은 절대 이름이어야 합니다. 만약 파일의 항목이 디렉토리일 경우, 하위 디렉토리까지 비교가 반복됩니다. 이 옵션 또는 **-t**를 사용해야 하며 동시에 둘 다 사용할 수는 없습니다.

-t 바이너리 파일이 아닌 파일만 비교합니다. 이 비교는 각 파일에 대해 `file(1)` 명령을 사용하여 해당 파일이 텍스트 파일인지 확인합니다. 이 옵션 또는 **-i**를 사용해야 하며 동시에 둘 다 사용할 수는 없습니다.

`-o outfile` 차이점의 출력을 `outfile`로 리디렉션합니다.
`BE_name` 활성 부트 환경과 비교할 부트 환경의 이름을 지정합니다.

예 7-2 부트 환경 비교

이 예에서 `first_disk` 부트 환경(소스)은 `second_disk` 부트 환경과 비교되며 결과는 파일로 보내집니다.

```
# /usr/sbin/lucompare -i /etc/lu/compare/ \  
-o /var/tmp/compare.out second_disk
```

비활성 부트 환경 삭제

`ludelete` 명령을 사용하여 부트 환경을 제거합니다. 다음 제한 사항을 확인합니다.

- 활성 부트 환경이나 다음 재부트에서 활성화되는 부트 환경은 삭제할 수 없습니다.
- 삭제할 부트 환경은 완료 상태여야 합니다. 완료된 부트 환경은 상태를 변경시킬 수 있는 작업에 참여하고 있지 않습니다. 부트 환경의 상태를 확인하려면 [126 페이지 “모든 부트 환경의 상태 표시”](#)를 사용합니다.
- `lumount`를 사용하여 마운트된 파일 시스템이 있는 부트 환경은 삭제할 수 없습니다.
- x86에만 해당: **Solaris 10 1/06 릴리스**부터 활성 GRUB 메뉴가 포함된 부트 환경을 삭제할 수 없습니다. 부트 환경을 다시 사용하려면 `lumake` 또는 `luupgrade` 명령을 사용합니다. 활성 GRUB 메뉴가 있는 부트 환경을 확인하려면 [System Administration Guide: Basic Administration](#)의 14 장, “Managing the Solaris Boot Archives (Tasks)”를 참조하십시오.

▼ 비활성 부트 환경 삭제

1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 다음을 입력합니다.

```
# ludelete BE_name  
  
BE_name     삭제할 비활성 부트 환경의 이름을 지정합니다.
```

예 7-3 비활성 부트 환경 삭제

이 예에서 부트 환경 `second_disk`가 삭제됩니다.

```
# ludelete second_disk
```

활성 부트 환경의 이름 표시

`lucurr` 명령을 사용하여 현재 실행 중인 부트 환경의 이름을 표시합니다. 시스템에 구성된 부트 환경이 없는 경우 "No Boot Environments are defined" 메시지가 표시됩니다. `lucurr`은 현재 부트 환경의 이름만 보고합니다. 다음에 재부트할 때 활성화될 부트 환경의 이름은 보고하지 않습니다. 부트 환경의 상태를 확인하려면 [126 페이지](#) “모든 부트 환경의 상태 표시”를 참조하십시오.

▼ 활성 부트 환경의 이름 표시

- 1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# /usr/sbin/lucurr
```

예 7-4 활성 부트 환경의 이름 표시

이 예에서는 현재 부트 환경의 이름이 표시됩니다.

```
# /usr/sbin/lucurr
solaris10
```

부트 환경의 이름 변경

Solaris 릴리스에서 다른 릴리스로 부트 환경을 업그레이드할 때 부트 환경의 이름을 변경하면 유용할 수 있습니다. 예를 들어 다음 운영 체제 업그레이드에서 부트 환경 `solaris8`의 이름을 `solaris10`으로 변경할 수 있습니다.

`lurename` 명령을 사용하여 비활성 부트 환경의 이름을 변경합니다.

x86 전용 - Solaris 10 1/06 릴리스부터 이름 바꾸기 또는 `lurename` 명령을 사용하면 GRUB 메뉴가 자동으로 업데이트됩니다. 업데이트된 GRUB 메뉴의 부트 항목 목록에 부트 환경의 이름이 표시됩니다. GRUB 메뉴에 대한 자세한 내용은 54 페이지 “다중 부트 환경 부트”를 참조하십시오.

GRUB 메뉴의 `menu.lst` 파일 위치를 확인하려면 [System Administration Guide: Basic Administration](#)의 14 장, “Managing the Solaris Boot Archives (Tasks)”를 참조하십시오.

표 7-2 부트 환경 이름 변경에 대한 제한

제한	수행 방법
이름의 길이는 30자를 넘지 않아야 합니다.	
이름은 영숫자와 UNIX 셸에만 국한되지 않는 기타 ASCII 문자로만 구성될 수 있습니다.	<code>sh(1)</code> 의 “할당” 부분을 참조하십시오.
이름에는 1바이트, 8비트 문자만 포함될 수 있습니다.	
이름은 시스템에서 고유해야 합니다.	
부트 환경의 이름을 바꾸기 전에 해당 부트 환경이 “완료” 상태여야 합니다.	부트 환경의 상태를 확인하려면 126 페이지 “모든 부트 환경의 상태 표시”를 참조하십시오.
<code>lumount</code> 또는 <code>mount</code> 를 사용하여 마운트된 파일 시스템이 있는 부트 환경의 이름은 바꿀 수 없습니다.	

▼ 비활성 부트 환경의 이름 변경

1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 다음을 입력합니다.

```
# lurename -e BE_name -n new_name
-e BE_name    변경할 비활성 부트 환경 이름을 지정합니다.
-n new_name   비활성 부트 환경의 새 이름을 지정합니다.
이 예에서는 second_disk의 이름이 third_disk로 변경됩니다.

# lurename -e second_disk -n third_disk
```

부트 환경 이름과 관련된 설명 추가 또는 변경

부트 환경 이름에 설명을 연관시킬 수 있습니다. 해당 설명은 결코 이름을 대체하지 않습니다. 부트 환경 이름은 길이와 문자에 제한이 따르지만 설명의 길이와 내용에는 제한이 없습니다. 설명은 단순 텍스트로 표현되거나 gif 파일처럼 복잡할 수 있습니다. 이런 설명을 다음과 같은 경우에 만들 수 있습니다.

- `lucreate` 명령을 사용하여 부트 환경을 만들고 `-A` 옵션을 사용할 경우
- `ludesc` 명령을 사용해 부트 환경을 만든 후

`lucreate`과 함께 `-A` 옵션을 사용하는 방법에 대한 [62 페이지 “처음으로 부트 환경 만들기”](#) 자세한 내용은

부트 환경을 만든 후 설명을 작성하는 방법에 대한 [ludesc\(1M\)](#) 자세한 내용은

▼ 텍스트를 사용하여 부트 환경 이름에 대한 설명 추가 또는 변경

- 1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# /usr/sbin/ludesc -n BE_name 'BE_description'
```

`-n BE_name 'BE_description'` 부트 환경 이름과 해당 이름에 연결할 새 설명을 지정합니다.

예 7-5 텍스트를 사용하여 부트 환경 이름에 대한 설명 추가

이 예에서 이름이 `second_disk`인 부트 환경에 부트 환경 설명이 추가됩니다. 설명은 작은 따옴표로 묶여 있는 텍스트입니다.

```
# /usr/sbin/ludesc -n second_disk 'Solaris 10 5/09 test build'
```

▼ 파일을 사용하여 부트 환경 이름에 대한 설명 추가 또는 변경

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# /usr/sbin/ludesc -n BE_name -f file_name
-n BE_name      부트 환경 이름을 지정합니다.
file_name       부트 환경에 연결할 파일을 지정합니다.
```

예 7-6 파일을 사용하여 부트 환경 이름에 대한 설명 추가

이 예에서 이름이 `second_disk`인 부트 환경에 부트 환경 설명이 추가됩니다. 설명은 `gif` 파일에 포함되어 있습니다.

```
# /usr/sbin/ludesc -n second_disk -f rose.gif
```

▼ 텍스트 설명을 사용하여 부트 환경 이름 결정

다음 명령은 지정한 설명과 연관된 부트 환경의 이름을 반환합니다.

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# /usr/sbin/ludesc -A 'BE_description'
-A 'BE_description'  부트 환경 이름과 연결할 설명을 지정합니다.
```

예 7-7 설명을 사용하여 부트 환경 이름 결정

이 예에서 `-A` 옵션을 설명과 함께 사용하여 부트 환경의 이름 `second_disk`를 확인합니다.

```
# /usr/sbin/ludesc -A 'Solaris 10 5/09 test build'
second_disk
```

▼ 파일의 설명을 사용하여 부트 환경 이름 결정

다음 명령은 파일과 연결된 부트 환경의 이름을 표시합니다. 파일에는 부트 환경에 대한 설명이 들어 있습니다.

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# /usr/sbin/ludesc -f file_name
```

-f *file_name* 부트 환경의 설명이 들어 있는 파일의 이름을 지정합니다.

예 7-8 파일의 설명을 사용하여 부트 환경 이름 결정

이 예에서는 -f 옵션과 설명을 포함하고 있는 파일의 이름을 사용하여 부트 환경의 이름인 `second_disk`를 결정합니다.

```
# /usr/sbin/ludesc -f rose.gif
second_disk
```

▼ 이름을 사용하여 부트 환경 설명 결정

이 절차에서는 명령에 지정된 부트 환경의 설명을 표시합니다.

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# /usr/sbin/ludesc -n BE_name
```

-n *BE_name* 부트 환경 이름을 지정합니다.

예 7-9 이름을 사용하여 부트 환경 설명 결정

이 예에서는 -n 옵션과 부트 환경 이름을 사용하여 설명을 결정합니다.

```
# /usr/sbin/ludesc -n second_disk
Solaris 10 5/09 test build
```

부트 환경의 구성 보기

lufslist 명령을 사용하여 부트 환경의 구성을 나열합니다. 출력에는 각 부트 환경 마운트 지점에 대한 디스크 슬라이스(파일 시스템), 파일 시스템 유형 및 파일 시스템 크기가 포함됩니다.

▼ 부트 환경의 구성 보기

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 다음을 입력합니다.

```
# lufslist -n BE_name
```

BE_name 파일 시스템 세부 사항을 보고자 하는 부트 환경의 이름을 지정합니다.

다음 예는 목록을 표시합니다.

Filesystem	fstype	size(Mb)	Mounted on
/dev/dsk/c0t0d0s1	swap	512.11	-
/dev/dsk/c0t4d0s3	ufs	3738.29	/
/dev/dsk/c0t4d0s4	ufs	510.24	/opt

주 - 비전역 영역이 포함된 목록의 예는 151 페이지 “부트 환경의 비전역 영역 파일 시스템의 구성 보기”를 참조하십시오.

비전역 영역이 설치된 시스템에서 Solaris OS 업그레이드

이 장에서는 Solaris Live Upgrade를 사용하여 비전역 영역이 설치된 시스템을 업그레이드하는 방법에 대해 설명합니다.

주 - 이 장에서는 UFS 파일 시스템용 Solaris Live Upgrade에 대해 설명합니다. 비전역 영역이 있는 UFS 파일 시스템을 ZFS 루트 폴로 마이그레이션하는 절차는 14 장, “비전역 영역이 설치된 ZFS용 Solaris Live Upgrade”를 참조하십시오.

이 장은 다음 절로 구성되어 있습니다.

- 비전역 영역이 포함된 시스템을 Solaris Live Upgrade로 업그레이드할 때 변경 사항에 대한 요약 내용은 137 페이지 “비전역 영역이 설치된 경우 Solaris Live Upgrade를 사용하여 업그레이드(개요)”를 참조하십시오.
- 새 부트 환경을 만들고 Solaris Live Upgrade로 시스템을 업그레이드하는 방법에 대한 단계별 지침은 144 페이지 “비전역 영역이 설치된 경우 부트 환경 만들기 및 업그레이드(작업)”를 참조하십시오.
- 새 부트 환경을 만들고 Solaris Live Upgrade로 시스템을 업그레이드하는 예와 간단한 지침에 대해서는 149 페이지 “비전역 영역이 설치된 시스템 업그레이드(예)”를 참조하십시오.
- 비전역 영역을 만드는 일반 정보에 대해서는 [System Administration Guide: Solaris Containers-Resource Management and Solaris Zones](#) 를 참조하십시오.

비전역 영역이 설치된 경우 Solaris Live Upgrade를 사용하여 업그레이드(개요)

Solaris Solaris 10 8/07 릴리스부터는 Solaris Live Upgrade를 사용하여 비전역 영역이 포함된 시스템을 업그레이드하거나 패치할 수 있습니다. 비전역 영역이 포함된 시스템이 있는 경우 권장되는 패치 업그레이드 및 추가 프로그램은 Solaris Live

Upgrade입니다. 다른 업그레이드 프로그램은 업그레이드를 완료하는 데 필요한 시간이 설치된 비전역 영역의 수에 따라 연속적으로 증가하므로 업그레이드에 상당한 시간이 걸릴 수 있습니다. Solaris Live Upgrade를 사용하여 시스템을 패치하면 단일 사용자 모드로 전환할 필요가 없으므로 시스템 가동 시간을 최대화할 수 있습니다. 다음 목록에는 비전역 영역이 설치된 시스템을 보관하기 위한 변경 사항이 요약되어 있습니다.

- 새로운 패키지인 SUNWlucfg를 다른 Solaris Live Upgrade 패키지인 SUNWlur 및 SUNWluu와 함께 설치해야 합니다. 이 패키지는 비전역 영역이 설치된 시스템뿐만 아니라 모든 시스템에 필요합니다.
- 현재 실행 중인 부트 환경에서 새 부트 환경을 만드는 방법은 한 가지 예외를 제외하고 이전 릴리스와 같습니다. 비전역 영역의 공유 파일 시스템에 대한 대상 디스크 슬라이스를 지정할 수 있습니다. 자세한 내용은 144 페이지 “비전역 영역이 설치된 경우 부트 환경 만들기 및 업그레이드(작업)”를 참조하십시오.
- lumount 명령을 사용하면 비전역 영역에서 비활성 부트 환경에 존재하는 해당 파일 시스템에 액세스할 수 있습니다. 전역 영역 관리자가 lumount 명령을 사용하여 비활성 부트 환경을 마운트하는 경우 부트 환경이 비전역 영역에 대해서도 마운트됩니다. 152 페이지 “비전역 영역이 포함된 시스템에서 lumount 명령 사용”를 참조하십시오.
- 부트 환경 비교 기능이 향상되었습니다. lucompare 명령은 이제 비전역 영역의 내용을 포함하는 부트 환경 비교를 생성합니다. 151 페이지 “비전역 영역이 설치된 시스템의 부트 환경 비교”를 참조하십시오.
- lufslist 명령을 사용한 파일 시스템 나열이 향상되어 전역 영역과 비전역 영역 모두에 대한 파일 시스템을 나열합니다. 151 페이지 “부트 환경의 비전역 영역 파일 시스템의 구성 보기”를 참조하십시오.

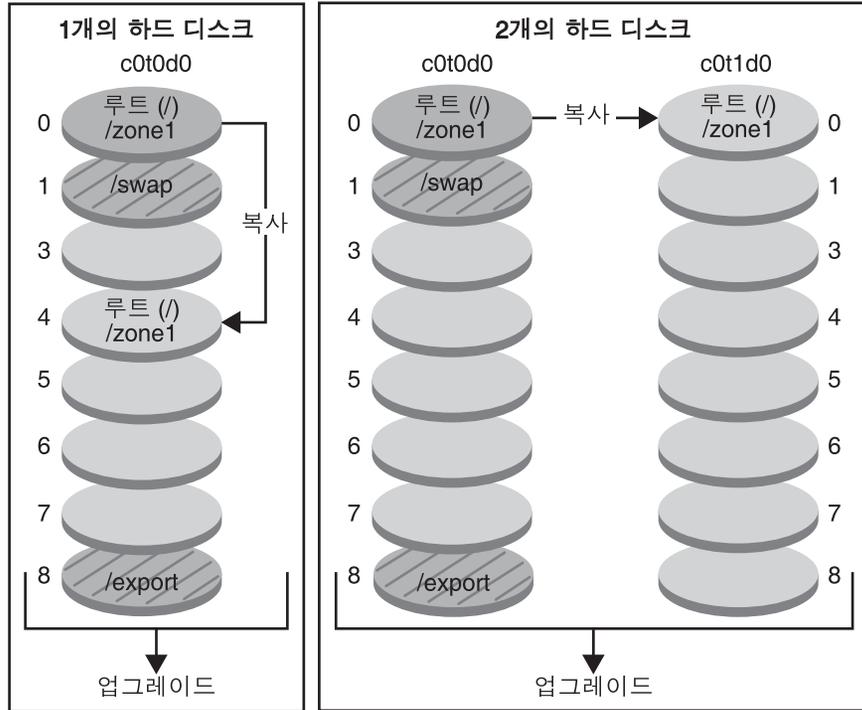
Solaris Zones 및 Solaris Live Upgrade 이해

Solaris 영역 분할 기술은 운영 체제 서비스를 가상화하고 실행 중인 응용 프로그램에 대해 격리되고 안전한 환경을 제공하는 데 사용됩니다. 비전역 영역은 Solaris OS의 단일 인스턴스(전역 영역) 내에 만들어진 가상화된 운영 체제 환경입니다. 비전역 영역을 작성하면 프로세스를 나머지 시스템으로부터 격리시키는 응용 프로그램 실행 환경을 만들게 됩니다.

Solaris Live Upgrade는 현재 실행 중인 시스템을 새 슬라이스로 복사하는 메커니즘입니다. 비전역 영역이 설치된 경우에는 전역 영역의 파일 시스템과 함께 비활성 부트 환경으로 복사될 수 있습니다.

그림 8-1에서는 전역 영역의 파일 시스템과 함께 비활성 부트 환경으로 복사된 비전역 영역을 보여 줍니다.

부트 환경 작성 - 비전역 영역 복사



1개의 디스크 명령:
 # lucreate -c bootenv1 \
 -m /dev/dsk/c0t0d0s4:ufs \
 -n bootenv2

2개의 디스크 명령:
 # lucreate -c bootenv1 \
 -m /dev/dsk/c0t1d0s0:ufs \
 -n bootenv2

- 현재 릴리스 X
중요한 파일 시스템 루트 (/)
- 비활성 릴리스 X
중요한 파일 시스템 루트 (/)
- 공유 파일 시스템

그림 8-1 부트 환경 만들기 - 비전역 영역 복사

- 단일 디스크가 있는 이 시스템의 예에서는 루트(/) 파일 시스템이 c0t0d0s4로 복사됩니다. 파일 시스템과 연관된 모든 비전역 영역이 s4로 복사됩니다. /export 파일 시스템과 /swap 볼륨은 현재 부트 환경 bootenv1과 비활성 부트 환경 bootenv2 간에 공유됩니다. lucreate 명령은 다음과 같습니다.

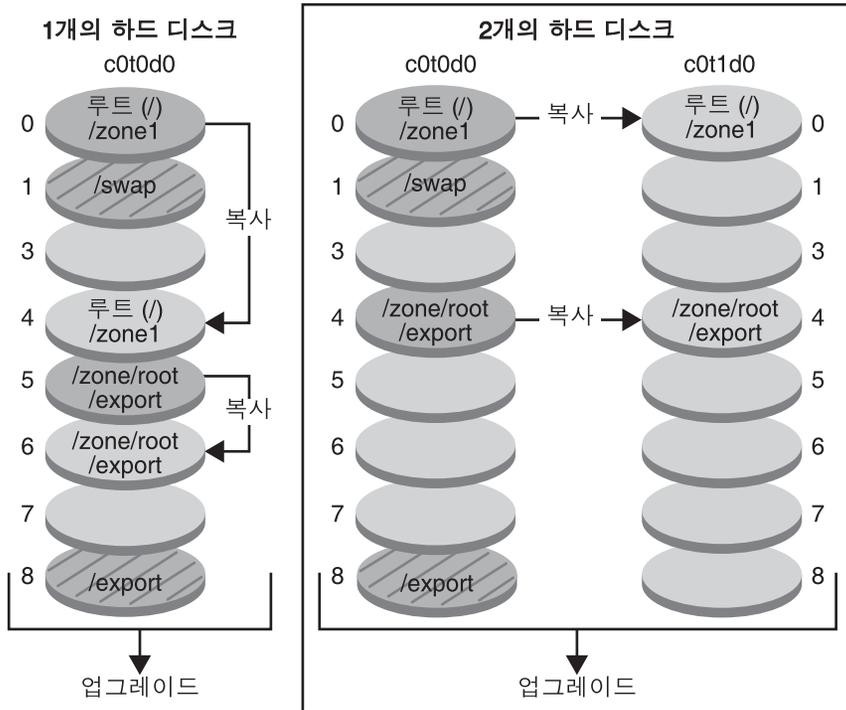
```
# lucreate -c bootenv1 -m /dev/dsk/c0t0d0s4:ufs -n bootenv2
```

- 두 개의 디스크가 있는 이 시스템의 예에서는 루트(/) 파일 시스템이 `c0t1d0s0`으로 복사됩니다. 파일 시스템과 연관된 모든 비전역 영역이 `s0`으로 복사됩니다. `/export` 파일 시스템과 `/swap` 볼륨은 현재 부트 환경 `bootenv1`과 비활성 부트 환경 `bootenv2` 간에 공유됩니다. `lucreate` 명령은 다음과 같습니다.

```
# lucreate -c bootenv1 -m /:/dev/dsk/c0t1d0s0:ufs -n bootenv2
```

그림 8-2에서는 비전역 영역이 비활성 부트 환경으로 복사되었음을 보여 줍니다.

부트 환경 작성 - 비전역 영역에서 공유 파일 시스템을 복사



1개의 디스크 명령:
 # lucreate -c bootenv1 \
 -m /dev/dsk/c0t0d0s4:ufs \
 -m /export:/dev/dsk/c0t0d0s6:ufs:zone1
 -n bootenv2

2개의 디스크 명령:
 # lucreate -c bootenv1 \
 -m /dev/dsk/c0t1d0s1:ufs \
 -m /export:/dev/dsk/c0t1d0s4:ufs:zone1
 -n bootenv2

- 현재 릴리스 X
중요한 파일 시스템 루트 (/)
- 비활성 릴리스 X
중요한 파일 시스템 루트 (/)
- 공유 파일 시스템

그림 8-2 부트 환경 만들기 - 비전역 영역에서 공유 파일 시스템을 복사

- 단일 디스크가 있는 이 시스템의 예에서는 루트(/) 파일 시스템이 c0t0d0s4로 복사됩니다. 파일 시스템과 연관된 모든 비전역 영역이 s4로 복사됩니다. 비전역 영역 zone1에는 zonecfg add fs 명령으로 만들어진 별도의 파일 시스템이 있습니다. 영역 경로는 /zone1/root/export입니다. 이 파일 시스템이 비활성 부트 환경에서 공유되지 않도록 파일 시스템은 별도의 슬라이스인 c0t0d0s6에 배치됩니다. /export 파일 시스템과 /swap 볼륨은 현재 부트 환경 bootenv1과 비활성 부트 환경 bootenv2 간에 공유됩니다. lucreate 명령은 다음과 같습니다.

```
# lucreate -c bootenv1 -m /:/dev/dsk/c0t0d0s4:ufs \
-m /export:/dev/dsk/c0t0d0s6:ufs:zone1 -n bootenv2
```

- 두 개의 디스크가 있는 이 시스템의 예에서는 루트(/) 파일 시스템이 c0t1d0s0으로 복사됩니다. 파일 시스템과 연관된 모든 비전역 영역이 s0으로 복사됩니다. 비전역 영역 zone1에는 zonecfg add fs 명령으로 만들어진 별도의 파일 시스템이 있습니다. 영역 경로는 /zone1/root/export입니다. 이 파일 시스템이 비활성 부트 환경에서 공유되지 않도록 파일 시스템은 별도의 슬라이스인 c0t1d0s4에 배치됩니다. /export 파일 시스템과 /swap 볼륨은 현재 부트 환경 bootenv1과 비활성 부트 환경 bootenv2 간에 공유됩니다. lucreate 명령은 다음과 같습니다.

```
# lucreate -c bootenv1 -m /:/dev/dsk/c0t1d0s0:ufs \
-m /export:/dev/dsk/c0t1d0s4:ufs:zone1 -n bootenv2
```

비전역 영역에서 Solaris Live Upgrade 사용을 위한 지침(계획)

비전역 영역 사용 계획에는 다음과 같은 제한 사항이 있습니다.

표 8-1 비전역 영역을 통해 업그레이드할 경우 제한 사항

문제	설명
영역이 설치된 시스템에서 Solaris Live Upgrade를 사용할 때 다음 사항을 고려하십시오. lucreate 및 lumount 작업 중에 영역 상태가 전환되지 않도록 해야 합니다.	<ul style="list-style-type: none"> ■ lucreate 명령을 사용하여 비활성 부트 환경을 만들 때 지정된 비전역 영역이 실행되고 있지 않을 경우 lucreate 작업이 완료될 때까지 영역을 부트할 수 없습니다. ■ lucreate 명령을 사용하여 비활성 부트 환경을 만들 때 지정된 비전역 영역이 실행 중인 경우 lucreate 작업이 완료될 때까지 영역을 중지하거나 재부트해서는 안 됩니다. ■ lumount 명령을 사용하여 비활성 부트 환경을 마운트할 경우 lumount 작업 이전에 실행 중이었던 영역을 계속 실행할 수 있지만 비전역 영역을 부트 또는 재부트할 수는 없습니다. ■ 비전역 영역은 비전역 영역 관리자와 전역 영역 관리자가 모두 제어할 수 있기 때문에 상호 작용을 방지하려면 lucreate 또는 lumount 작업 중에 모든 영역을 중지하십시오.

표 8-1 비전역 영역을 통해 업그레이드할 경우 제한 사항 (계속)

문제	설명
전역 영역 관리자가 비전역 영역 관리자에게 Solaris Live Upgrade를 통한 업그레이드를 알리지 않을 경우 문제가 발생할 수 있습니다.	<p>Solaris Live Upgrade 작업이 진행 중일 경우 비전역 영역 관리자도 반드시 참여해야 합니다. 업그레이드는 업그레이드로 인해 발생하는 변경 사항을 해결할 관리자의 작업에 영향을 줍니다. 영역 관리자는 로컬 패키지가 시퀀스 진행 중 안정적인지 확인하고 구성 파일 조정과 같은 사후 업그레이드 작업을 처리하며 시스템 중단 일정을 수립해야 합니다.</p> <p>예를 들어, 전역 영역 관리자가 <code>lucreate</code> 명령을 사용하여 파일 시스템을 복사하는 동안 비전역 영역 관리자가 패키지를 추가하면 새 패키지가 파일 시스템과 함께 복사되지 않고 비전역 영역 관리자는 이 문제를 인식하지 못합니다.</p>

비전역 영역이 별도의 파일 시스템에 있을 때 부트 환경 만들기

현재 실행 중인 부트 환경에서 새 부트 환경을 만드는 작업은 한 가지 예외를 제외하고 이전 릴리스와 동일합니다. 비전역 영역의 공유 파일 시스템에 대한 대상 디스크 슬라이스를 지정할 수 있습니다. 이 예외는 다음 조건에서 발생합니다.

- 현재 부트 환경에서 `zonecfg add fs` 명령을 사용하여 비전역 영역에 대한 별도의 파일 시스템을 만든 경우
- 이 개별 파일 시스템이 공유 파일 시스템(예: `/zone/root/export`)에 있는 경우

별도의 파일 시스템이 새 부트 환경에서 공유되지 않도록 하려면 `lucreate` 명령을 사용하여 비전역 영역에 대한 별도의 파일 시스템의 대상 슬라이스를 지정할 수 있습니다. `-m` 옵션의 인수에 새 옵션 필드인 `zonename`이 제공됩니다. 이 새로운 필드는 비전역 영역의 개별 파일 시스템을 새 부트 환경의 개별 슬라이스에 배치합니다. 별도의 파일 시스템을 사용하여 비전역 영역을 설정하는 방법에 대한 자세한 내용은 `zonecfg(1M)`를 참조하십시오.

주 - 기본적으로 중요 파일 시스템(루트(/), `/usr` 및 `/opt` 파일 시스템)을 제외한 모든 파일 시스템은 현재 부트 환경과 새 부트 환경 간에 공유됩니다. 활성 부트 환경에서 공유된 파일을 업데이트하면 비활성 부트 환경에서도 데이터가 업데이트됩니다. 예를 들어, `/export` 파일 시스템은 공유 파일 시스템입니다. `-m` 옵션과 `zonename` 옵션을 사용하면 비전역 영역의 파일 시스템이 별도의 슬라이스로 복사되고 데이터가 공유되지 않습니다. 이 옵션을 사용하면 `zonecfg add fs` 명령을 사용하여 만든 비전역 영역 파일 시스템이 부트 환경 간에 공유되지 않습니다.

비전역 영역이 설치된 경우 부트 환경 만들기 및 업그레이드(작업)

다음 절에서는 비전역 영역이 설치된 경우의 단계별 업그레이드 절차에 대해 설명합니다.

- 144 페이지 “시스템에 비전역 영역이 설치된 경우 Solaris Live Upgrade로 업그레이드(작업)”
간략한 절차와 예에 대해서는 149 페이지 “비전역 영역이 설치된 시스템 업그레이드(예)”를 참조하십시오.

▼ 시스템에 비전역 영역이 설치된 경우 Solaris Live Upgrade로 업그레이드(작업)

다음 절차에서는 비전역 영역이 설치된 시스템에서 Solaris Live Upgrade를 사용하여 업그레이드하는 방법에 대한 자세한 지침을 제공합니다.

- 1 **Solaris Live Upgrade**를 처음으로 실행하기 전에 설치 매체에서 최신 **Solaris Live Upgrade** 패키지를 설치하고 **SunSolve** 정보 설명서 206844에 나열된 패치를 설치해야 합니다. **SunSolve** 웹 사이트에서 정보 설명서 206844(이전의 72099)를 검색합니다.

최신 패키지 및 패치를 설치하면 릴리스에 모든 최신 버그 수정과 새로운 기능이 포함됩니다. 새 부트 환경 만들기를 진행하기 전에 시스템과 관련된 모든 패치를 설치해야 합니다.

다음 하위 단계에서는 SunSolve 정보 설명서 206844에 나오는 단계에 대해 설명합니다.

- a. 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
- b. **Solaris Live Upgrade** 패키지를 제거하거나 추가하려면 SunSolve 웹 사이트에서 제공되는 정보 설명서 206844의 지침을 따르십시오.

다음 지침은 패키지를 제거 및 추가하는 정보 설명서 단계를 요약한 것입니다.

- 기존 Solaris Live Upgrade 패키지를 제거합니다.

세 개의 Solaris Live Upgrade 패키지 SUNWluu, SUNWlur 및 SUNWlucfg는 Solaris Live Upgrade를 사용하여 업그레이드 또는 패치하기 위해 필요한 소프트웨어로 구성됩니다. 이들 패키지는 기존 소프트웨어, 새 기능, 버그 수정 등을 포함합니다. Solaris Live Upgrade를 사용하기 전에 시스템에서 기존 패키지를 제거하지 않고 새 패키지를 설치한 경우에는 대상 릴리스로 업그레이드 또는 패치할 수 없습니다. SUNWlucfg 패키지는 **Solaris 10 8/07 릴리스부터** 지원되는 새로운 기능입니다. Solaris 10 8/07 이전 릴리스의 Solaris Live Upgrade 패키지를 사용하는 경우에는 이 패키지를 제거하지 않아도 됩니다.

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- 새 Solaris Live Upgrade 패키지를 설치합니다.

설치 DVD 또는 CD에 있는 liveupgrade20 명령을 사용하여 패키지를 설치할 수 있습니다. liveupgrade20 명령을 사용하려면 Java 소프트웨어가 필요합니다. 시스템에 Java 소프트웨어를 설치하지 않은 경우 pkgadd 명령을 사용하여 패키지를 설치해야 합니다. 자세한 내용은 SunSolve 정보 설명서를 참조하십시오.

- Solaris 운영 체제 DVD를 사용할 경우 디렉토리를 변경하고 설치 프로그램을 실행합니다.
 - 디렉토리를 변경합니다.

```
# cd /cdrom/cdrom0/Solaris_10/Tools/Installers
```

주 - SPARC 기반 시스템의 경우 설치 프로그램 경로가 Solaris 10 10/08 이전 릴리스와 다릅니다.

```
# cd /cdrom/cdrom0/s0/Solaris_10/Tools/Installers
```

- 설치 프로그램 실행

```
# ./liveupgrade20
```

Solaris 설치 프로그램 GUI가 표시됩니다. 스크립트를 사용할 경우 -noconsole 및 -nodisplay 옵션을 사용하여 GUI가 표시되지 않게 할 수 있습니다.

- Solaris Software - 2 CD를 사용하는 경우 경로를 변경하지 않고 설치 프로그램을 실행할 수 있습니다.

```
% ./installer
```

- 패키지가 성공적으로 설치되었는지 확인합니다.

```
# pkgchk -v SUNWLucfg SUNWLur SUNWLu
```

c. 패치를 로컬 디스크에 저장할 경우 /var/tmp/lupatches와 같은 디렉토리를 만듭니다.

d. SunSolve 웹 사이트에서 패치 목록을 확인합니다.

e. 이 예에서처럼 패치 디렉토리를 변경합니다.

```
# cd /var/tmp/lupatches
```

f. 패치를 설치합니다.

```
# patchadd -M path-to-patchespatch-id patch-id
```

*path-to-patches*는 패치 디렉토리의 경로(예: /var/tmp/lupatches)입니다. *patch-id*는 패치 번호입니다. 공백을 사용하여 여러 패치 이름을 구분합니다.

주 - 정보 설명서 206844에 지정된 순서대로 패치를 적용해야 합니다.

g. 필요한 경우 시스템을 재부트합니다. 특정 패치는 재부트해야만 적용됩니다.

x86에만 해당: 시스템을 재부트해야 합니다. 그렇지 않으면 Solaris Live Upgrade가 실패합니다.

```
# init 6
```

이제 새 부트 환경을 성공적으로 만드는 데 필요한 패키지 및 패치가 설치되었습니다.

2 새 부트 환경을 만듭니다.

```
# lucreate [-A 'BE_description'] [-c BE_name] \
  -m mountpoint:device[,metadevice]:fs_options[:zonename] [-m ...] -n BE_name
```

-n *BE_name*

생성될 부트 환경의 이름 *BE_name*은 시스템에서 고유해야 합니다.

-A '*BE_description*'

(선택 사항) 부트 환경 이름(*BE_name*)과 연관된 부트 환경 설명을 작성할 수 있도록 합니다. 설명은 길이 제한이 없고 모든 문자를 포함할 수 있습니다.

-c *BE_name*

활성 부트 환경에 이름 *BE_name*을 할당합니다. 이 옵션은 필수 사항이 아니며 부트 환경을 처음 만들 때만 사용됩니다. 처음 *lucreate*를 실행할 때 -c 옵션을 생략하면 소프트웨어에서는 기본 파일 이름을 사용합니다.

-m *mountpoint:device[,metadevice]:fs_options[:zonename] [-m ...]*

*vfstab*에 새 부트 환경의 파일 시스템 구성을 지정합니다. -m에 대한 인자로 지정된 파일 시스템은 동일한 디스크에 있거나 여러 디스크에 분산될 수 있습니다. 이 옵션을 필요한 만큼 여러 번 사용하여 필요한 수의 파일 시스템을 만듭니다.

- *mountpoint*는 모든 유효한 마운트 지점 또는 -(하이픈)일 수 있으며 스왑 분할 영역을 나타냅니다.

- *device* 필드는 다음 중 하나일 수 있습니다.
 - */dev/dsk/cwtxdysz* 형식의 디스크 장치 이름
 - */dev/md/dsk/dnum* 형식의 Solaris 볼륨 관리자 볼륨 이름
 - */dev/md/vxfs/dsk/dnum* 형식의 Veritas 볼륨 관리자 볼륨 이름
 - 키워드 *merged*는 지정된 마운트 지점에 있는 파일 시스템이 부모와 병합될 것임을 나타냄.
- *fs_options* 필드는 다음 중 하나일 수 있습니다.
 - *ufs*, UFS 파일 시스템을 나타냄.
 - *vxfs*, Veritas 파일 시스템을 나타냄.
 - *swap*, 스왑 볼륨을 나타냅니다. 스왑 마운트 지점은 -(하이픈)이어야 합니다.
 - 로컬 장치(미러)인 파일 시스템의 경우 여러 가지 키워드로 이 파일 시스템에 적용할 작업을 지정합니다. 이러한 키워드는 논리적 장치를 만들거나 구성을 변경하거나 삭제할 수 있습니다. 이들 키워드에 대한 설명은 77 페이지 “RAID-1 볼륨(미러)을 사용하여 부트 환경 만들기”을 참조하십시오.
 - *zonename*은 비전역 영역에 있는 별도의 파일 시스템이 별도의 슬라이스에 배치되도록 지정합니다. 이 옵션은 해당 영역에 있는 별도의 파일 시스템이 */zone1/root/export*와 같은 공유 파일 시스템에 있는 경우 사용됩니다. 이 옵션은 해당 영역에 있는 별도의 파일 시스템을 새 슬라이스로 복사하고 이 파일 시스템이 공유되지 않도록 합니다. 별도의 파일 시스템이 *zonecfg add fs* 명령으로 만들어졌습니다.

다음 예에서는 *newbe*라는 새 부트 환경이 만들어집니다. 루트(/) 파일 시스템은 *c0t1d0s4*에 배치됩니다. 현재 부트 환경의 모든 비전역 영역이 새 부트 환경으로 복사됩니다. *zone1*이라는 비전역 영역에는 *c0t1d0s1*에서 별도의 마운트 지점이 지정됩니다.

주 - 기본적으로 중요 파일 시스템(루트(/), */usr* 및 */opt* 파일 시스템)을 제외한 모든 파일 시스템은 현재 부트 환경과 새 부트 환경 간에 공유됩니다. */export* 파일 시스템은 공유 파일 시스템입니다. *-m* 옵션을 사용하면 비전역 영역의 파일 시스템이 별도의 슬라이스에 배치되고 데이터가 공유되지 않습니다. 이 옵션을 사용하면 *zonecfg add fs* 명령으로 만들어진 영역 파일 시스템이 부트 환경 간에 공유되지 않습니다. 자세한 내용은 *zonecfg(1M)*를 참조하십시오.

```
# lucreate -n newbe -m /:/dev/dsk/c0t1d0s4:ufs -m /export:/dev/dsk/c0t1d0s1:ufs:zone1
```

3 부트 환경을 업그레이드합니다.

업그레이드하기 위해 사용할 운영 체제 이미지를 네트워크에서 가져옵니다.

```
# luupgrade -u -n BE_name -s os_image_path
```

-u 부트 환경에서 운영 체제 이미지를 업그레이드합니다.

-n *BE_name* 업그레이드될 부트 환경의 이름을 지정합니다.
 -s *os_image_path* 운영 체제 이미지를 포함하는 디렉토리의 경로 이름을 지정합니다.
 이 예에서는 새 부트 환경 *newbe*가 네트워크 설치 이미지에서 업그레이드됩니다.

```
# luupgrade -n newbe -u -s /net/server/export/Solaris_10/combined.solaris_wos
```

4 (선택 사항) 부트 환경이 부트 가능한지 확인합니다.

lustatus 명령은 부트 환경 만들기가 완료되었고 부트 가능한지 보고합니다.

```
# lustatus
boot environment  Is      Active  Active  Can      Copy
Name              Complete Now     OnReboot Delete  Status
-----
c0t1d0s0          yes     yes     yes     no      -
newbe             yes     no      no      yes     -
```

5 새 부트 환경을 활성화합니다.

```
# luactivate BE_name
```

*BE_name*은 활성화할 부트 환경의 이름을 지정합니다.

주 - x86 기반 시스템의 경우 luactivate 명령은 처음 부트 환경을 부트할 때 필요합니다. 이후 활성화는 GRUB 메뉴에서 부트 환경을 선택하여 수행할 수 있습니다. 단계별 지침에 대해서는 111 페이지 “x86: GRUB 메뉴를 사용하여 부트 환경 활성화”를 참조하십시오.

부트 환경을 성공적으로 활성화하려면 부트 환경이 몇 가지 조건을 충족해야 합니다. 자세한 내용은 107 페이지 “부트 환경 활성화”를 참조하십시오.

6 다시 부트합니다.

```
# init 6
```



주의 - init 또는 shutdown 명령만 사용하여 재부트합니다. reboot, halt 또는 uadmin 명령을 사용하면 시스템이 부트 환경을 전환하지 않습니다. 가장 최근의 활성 부트 환경이 다시 부트됩니다.

부트 환경이 전환되고 이제 새 부트 환경이 현재 부트 환경이 됩니다.

7 (선택 사항) 다른 부트 환경으로 폴백합니다.

새 부트 환경을 사용할 수 없거나 다른 부트 환경으로 전환하려면 6 장, “오류 복구: 원래 부트 환경으로 폴백(작업)”을 참조하십시오.

비전역 영역이 설치된 시스템 업그레이드(예)

다음 절차에서는 Solaris Live Upgrade로 업그레이드하기 위한 간략한 지침이 예와 함께 제공됩니다.

단계에 대한 자세한 내용은 144 페이지 “시스템에 비전역 영역이 설치된 경우 Solaris Live Upgrade로 업그레이드(작업)”를 참조하십시오.

시스템에 비전역 영역이 설치된 경우 Solaris Live Upgrade로 업그레이드

다음 예에서는 비전역 영역이 설치된 시스템을 업그레이드하기 위한 단계에 대해 간략하게 설명합니다. 이 예에서는 Solaris 10 릴리스가 실행 중인 시스템에서 `lucreate` 명령을 사용하여 새 부트 환경을 만듭니다. 이 시스템에는 비전역 영역이 설치되어 있고 공유 파일 시스템 `zone1/root/export`에 있는 별도의 파일 시스템을 사용하는 비전역 영역이 있습니다. 새 부트 환경은 `luupgrade` 명령을 사용하여 Solaris 10 5/09 릴리스로 업그레이드됩니다. 업그레이드된 부트 환경은 `luactivate` 명령을 사용하여 활성화됩니다.

주 - 이 절차는 시스템이 **볼륨 관리자**를 실행 중이라고 가정합니다. 볼륨 관리자를 사용하여 이동식 매체를 관리하는 방법에 대한 자세한 내용은 **System Administration Guide: Devices and File Systems**를 참조하십시오.

1. 필요한 패치를 설치합니다.

<http://sunsolve.sun.com>을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 206844(이전의 72099)를 검색합니다. 이 예에서 `/net/server/export/patches`는 패치 경로입니다.

```
# patchadd /net/server/export/patches
# init 6
```

2. 현재 부트 환경에서 Solaris Live Upgrade 패키지를 제거합니다.

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

3. Solaris DVD 또는 CD를 넣습니다. 그런 다음 대상 릴리스에서 대체 Solaris Live Upgrade 패키지를 설치합니다.

```
# pkgadd -d /cdrom/cdrom0/Solaris_10/Product SUNWlucfg SUNWlur SUNWluu
```

4. 부트 환경을 만듭니다.

다음 예에서는 newbe라는 새 부트 환경이 만들어집니다. 루트(/) 파일 시스템은 c0t1d0s4에 배치됩니다. 현재 부트 환경의 모든 비전역 영역이 새 부트 환경으로 복사됩니다. zone1에 대한 별도의 파일 시스템이 zonecfg add fs 명령으로 만들어졌습니다. 이 별도의 파일 시스템 /zone/root/export는 별도의 파일 시스템 c0t1d0s1에 배치됩니다. 이 옵션을 사용하면 별도의 파일 시스템이 현재 부트 환경과 새 부트 환경 간에 공유되지 않습니다.

```
# lucreate -n newbe -m /:/dev/dsk/c0t1d0s4:ufs -m /export:/dev/dsk/c0t1d0s1:ufs:zone1
```

5. 새 부트 환경을 업그레이드합니다.

이 예에서 /net/server/export/Solaris_10/combined.solaris_wos는 네트워크 설치 이미지의 경로입니다.

```
# luupgrade -n newbe -u -s /net/server/export/Solaris_10/combined.solaris_wos
```

6. (선택 사항) 부트 환경이 부트 가능한지 확인합니다.

lustatus 명령은 부트 환경 만들기의 완료 여부를 보고합니다.

```
# lustatus
```

boot environment Name	Is Complete	Active Now	Active OnReboot	Can Delete	Copy Status
c0t1d0s0	yes	yes	yes	no	-
newbe	yes	no	no	yes	-

7. 새 부트 환경을 활성화합니다.

```
# luactivate newbe
```

```
# init 6
```

이제 부트 환경 newbe가 활성화됩니다.

8. (선택 사항) 서로 다른 부트 환경으로 폴백 새 부트 환경을 사용할 수 없거나 다른 부트 환경으로 전환하려면 6장, “오류 복구: 원래 부트 환경으로 폴백(작업)”을 참조하십시오.

비전역 영역이 포함된 부트 환경 관리

다음 절에서는 비전역 영역이 포함된 부트 환경을 관리하는 방법에 대해 설명합니다.

▼ 부트 환경의 비전역 영역 파일 시스템의 구성 보기

이 절차에서는 전역 영역과 비전역 영역 모두의 파일 시스템 목록을 표시합니다.

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 파일 시스템의 목록을 표시합니다.

```
# lufslslist -n BE_name
```

BE_name 파일 시스템 세부 사항을 보고자 하는 부트 환경의 이름을 지정합니다.

예 8-1 비전역 영역이 있는 파일 시스템 나열

다음 예에서는 비전역 영역이 포함된 파일 시스템의 목록을 표시합니다.

```
# lufslslist -n s3
boot environment name: s3
This boot environment is currently active.
This boot environment will be active on next system boot.

Filesystem                fstype    device size Mounted on Mount Options
-----
/dev/dsk/c0t0d0s1         swap      2151776256 - -
/dev/dsk/c0t0d0s3         ufs       10738040832 / -
/dev/dsk/c0t0d0s7         ufs       10487955456 /export -
                        zone <zone1> within boot environment <s3>
/dev/dsk/c0t0d0s5         ufs       5116329984 /export -
```

▼ 비전역 영역이 설치된 시스템의 부트 환경 비교

이제 `lucompare` 명령은 비전역 영역의 내용이 포함된 부트 환경의 비교를 생성합니다.

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 현재 부트 환경과 새 부트 환경을 비교합니다.

```
# /usr/sbin/lucompare -i infile (or) -t -o outfile BE_name
```

- i *infile* *infile*에 나열된 파일들을 비교합니다. 비교할 파일의 이름은 절대 이름이어야 합니다. 파일의 항목이 디렉토리인 경우 비교가 디렉토리까지 반복됩니다. 이 옵션 또는 -t를 사용해야 하며 동시에 둘 다 사용할 수는 없습니다.
- t 바이너리 파일이 아닌 파일만 비교합니다. 이 비교는 각 파일에 대해 `file(1)` 명령을 사용하여 해당 파일이 텍스트 파일인지 확인합니다. 이 옵션 또는 -i를 사용해야 하며 동시에 둘 다 사용할 수는 없습니다.
- o *outfile* 차이점의 출력을 *outfile*로 리디렉션합니다.
- BE_name* 활성 부트 환경과 비교할 부트 환경의 이름을 지정합니다.

예 8-2 부트 환경 비교

이 예에서는 현재 부트 환경(소스)을 `second_disk` 부트 환경과 비교하고 그 결과를 파일로 보냅니다.

```
# /usr/sbin/lucompare -i /etc/lu/compare/ -o /var/tmp/compare.out second_disk
```

비전역 영역이 포함된 시스템에서 lumount 명령 사용

`lumount` 명령은 비활성 부트 환경에 있는 해당 파일 시스템에 대한 액세스 권한을 비전역 영역에 제공합니다. 전역 영역 관리자가 `lumount` 명령을 사용하여 비활성 부트 환경을 마운트하는 경우 부트 환경이 비전역 영역에 대해서도 마운트됩니다.

다음 예에서는 해당 파일 시스템이 전역 영역에 있는 `/mnt`에서 부트 환경 `newbe`에 마운트됩니다. 실행 중이거나 마운트되었거나, 준비된 비전역 영역의 경우 `newbe` 내의 해당 파일 시스템도 각 영역 내의 `/mnt`에서 사용할 수 있게 됩니다.

```
# lumount -n newbe /mnt
```

마운트에 대한 자세한 내용은 `lumount(1M)` 매뉴얼 페이지를 참조하십시오.

Solaris Live Upgrade (예)

이 장에서는 부트 환경을 만들고 새 부트 환경을 업그레이드 및 활성화하여 현재 실행 중인 시스템으로 만드는 예를 소개합니다.

주 - 이 장에서는 UFS 파일 시스템용 Solaris Live Upgrade에 대해 설명합니다. UFS 파일 시스템을 ZFS 루트 폴로 마이그레이션하거나 ZFS 루트 폴을 만들고 설치하는 절차에 대한 자세한 내용은 13 장, “ZFS 루트 폴의 부트 환경 만들기”를 참조하십시오.

이 장은 다음 절로 구성되어 있습니다.

- 153 페이지 “Solaris Live Upgrade를 사용한 업그레이드의 예”
- 161 페이지 “RAID-1 볼륨의 한쪽 분리 및 업그레이드 예(미러)”
- 165 페이지 “기존 볼륨에서 Solaris 볼륨 관리자 RAID-1 볼륨으로의 마이그레이션 예”
- 165 페이지 “빈 부트 환경 만들기 및 Solaris Flash 아카이브 설치 예”

Solaris Live Upgrade를 사용한 업그레이드의 예

이 예에서는 Solaris 9 릴리스가 실행되고 있는 시스템에서 `lucreate` 명령을 사용하여 새 부트 환경을 만듭니다. 새 부트 환경은 `luupgrade` 명령을 사용하여 Solaris 10 5/09 릴리스로 업그레이드됩니다. 업그레이드된 부트 환경은 `luactivate` 명령을 사용하여 활성화됩니다. 원래의 부트 환경으로 복구하는 예도 제공합니다.

Solaris Live Upgrade 사용 준비

Solaris Live Upgrade를 처음으로 실행하기 전에 설치 매체에서 최신 Solaris Live Upgrade 패키지를 설치하고 SunSolve 정보 설명서 206844에 나열된 패치를 설치해야 합니다. SunSolve 웹 사이트에서 정보 설명서 206844(이전의 72099)를 검색합니다.

최신 패키지 및 패치를 설치하면 릴리스에 모든 최신 버그 수정과 새로운 기능이 포함됩니다. 새 부트 환경 만들기를 진행하기 전에 시스템과 관련된 모든 패치를 설치해야 합니다.

다음 단계에서는 SunSolve 정보 설명서 206844에 나오는 단계에 대해 설명합니다.

주 - 이 절차는 시스템이 **볼륨 관리자**를 실행 중이라고 가정합니다. 볼륨 관리자를 사용하여 이동식 매체를 관리하는 방법에 대한 자세한 내용은 [System Administration Guide: Devices and File Systems](#)를 참조하십시오.

1. 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
2. Solaris Live Upgrade 패키지를 제거하거나 추가하려면 SunSolve 웹 사이트에서 제공되는 정보 설명서 206844의 지침을 따르십시오.

- a. 기존 Solaris Live Upgrade 패키지를 제거합니다.

세 개의 Solaris Live Upgrade 패키지 SUNWluu, SUNWlur 및 SUNWlucfg는 Solaris Live Upgrade를 사용하여 업그레이드 또는 패치하기 위해 필요한 소프트웨어로 구성됩니다. 이들 패키지는 기존 소프트웨어, 새 기능, 버그 수정 등을 포함합니다. Solaris Live Upgrade를 사용하기 전에 시스템에서 기존 패키지를 제거하지 않고 새 패키지를 설치한 경우에는 대상 릴리스로 업그레이드 또는 패치할 수 없습니다. SUNWlucfg 패키지는 **Solaris 10 8/07 릴리스부터** 지원되는 새로운 기능입니다. Solaris 10 8/07 이전 릴리스의 Solaris Live Upgrade 패키지를 사용하는 경우에는 이 패키지를 제거하지 않아도 됩니다.

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- b. 새 Solaris Live Upgrade 패키지를 설치합니다.

설치 DVD 또는 CD에 있는 liveupgrade20 명령을 사용하거나 pkgadd 명령을 사용하여 패키지를 설치할 수 있습니다. liveupgrade20 명령을 사용하려면 Java 소프트웨어가 필요합니다. 시스템에 Java 소프트웨어를 설치하지 않은 경우 pkgadd 명령을 사용하여 패키지를 설치해야 합니다. 자세한 내용은 SunSolve 정보 설명서를 참조하십시오.

- Solaris 운영 체제 DVD를 사용할 경우 디렉토리를 변경하고 설치 프로그램을 실행합니다.
- 디렉토리를 변경합니다.

```
# cd /cdrom/cdrom0/Solaris_10/Tools/Installers
```

주 - SPARC 기반 시스템의 경우 설치 프로그램 경로가 Solaris 10 10/08 이전 릴리스와 다릅니다.

```
# cd /cdrom/cdrom0/s0/Solaris_10/Tools/Installers
```

- 설치 프로그램 실행

```
# ./liveupgrade20 -noconsole -nodisplay
```

-noconsole 및 -nodisplay 옵션은 문자 사용자 인터페이스(CUI)가 표시되지 않게 합니다.

주 - Solaris Live Upgrade CUI는 더 이상 지원되지 않습니다.

- Solaris Software - 2 CD를 사용하는 경우 경로를 변경하지 않고 설치 프로그램을 실행할 수 있습니다.

```
% ./installer
```

- 패키지가 성공적으로 설치되었는지 확인합니다.

```
# pkgchk -v SUNWLucfg SUNWLur SUNWLu
```

3. 정보 설명서 206844에 나열된 패치를 설치합니다.

- a. 패치를 로컬 디스크에 저장할 경우 /var/tmp/lupatches와 같은 디렉토리를 만듭니다.
- b. SunSolve 웹 사이트에서 패치 목록을 확인합니다.
- c. 이 예에서처럼 패치 디렉토리를 변경합니다.

```
# cd /var/tmp/lupatches
```

- d. 패치를 설치합니다.

```
# patchadd -M path-to-patchespatch-id patch-id
```

*path-to-patches*는 패치 디렉토리의 경로(예: /var/tmp/lupatches)입니다.
*patch-id*는 패치 번호입니다. 공백을 사용하여 여러 패치 이름을 구분합니다.

주 - 정보 설명서 206844에 지정된 순서대로 패치를 적용해야 합니다.

- e. 필요한 경우 시스템을 재부트합니다. 특정 패치는 재부트해야만 적용됩니다.
x86에만 해당: 시스템을 재부트해야 합니다. 그렇지 않으면 Solaris Live Upgrade가 실패합니다.

```
# init 6
```

이제 새 부트 환경을 성공적으로 만드는 데 필요한 패키지 및 패치가 설치되었습니다.

부트 환경 만들기

소스 부트 환경의 이름은 `-c` 옵션을 사용하여 `c0t4d0s0`으로 지정됩니다. 처음으로 부트 환경을 만들 때만 소스 부트 환경의 이름을 지정합니다. `-c` 옵션을 사용하여 이름을 지정하는 방법은 “처음으로 부트 환경 만들기” [단계 2](#)의 설명을 참조하십시오.

새 부트 환경 이름은 `c0t15d0s0`으로 지정됩니다. `-A` 옵션은 부트 환경 이름과 관련된 설명을 작성합니다.

루트(/) 파일 시스템은 새 부트 환경으로 복사됩니다. 또한 소스 부트 환경의 스왑 슬라이스를 공유하는 것이 아니라 새 스왑 슬라이스가 생성됩니다.

```
# lucreate -A 'BE_description' -c /dev/dsk/c0t4d0s0 -m /:/dev/dsk/c0t15d0s0:ufs\
-m -:/dev/dsk/c0t15d0s1:swap -n /dev/dsk/c0t15d0s0
```

비활성 부트 환경 업그레이드

비활성 부트 환경의 이름은 `c0t15d0s0`으로 지정됩니다. 업그레이드하기 위해 사용할 운영 체제 이미지를 네트워크에서 가져옵니다.

```
# luupgrade -n c0t15d0s0 -u -s /net/ins-svr/export/Solaris_10 \
combined.solaris_wos
```

부트 환경의 부트 가능 여부 검사

`lustatus` 명령은 부트 환경 만들기의 완료 여부를 보고합니다. `lustatus`는 부트 환경의 부트 가능성도 보고합니다.

```
# lustatus
boot environment  Is      Active  Active   Can      Copy
Name             Complete Now      OnReboot Delete   Status
-----
c0t4d0s0         yes     yes     yes     no      -
c0t15d0s0         yes     no      no      yes     -
```

비활성 부트 환경 활성화

`c0t15d0s0` 부트 환경은 `luactivate` 명령을 통해 부트 가능 상태가 됩니다. 이렇게 하면 시스템이 재부트되며 `c0t15d0s0`이 활성 부트 환경이 됩니다. 이제 `c0t4d0s0` 부트 환경은 사용하지 않게 됩니다.

```
# luactivate c0t15d0s0
# init 6
```

(선택 사항) 소스 부트 환경으로 폴백

폴백 절차에는 새 부트 환경 활성화 상태에 따라 다음 세 가지가 있습니다.

- SPARC 기반 시스템의 경우:
 - 활성화는 성공적이지만 원래 부트 환경으로 복귀하고자 합니다. [예 9-1](#)을 참조하십시오.
 - 활성화에 실패했으며 원래의 부트 환경으로 부트할 수 있습니다. [예 9-2](#)를 참조하십시오.
 - 활성화에 실패했으며 매체나 네트워크 설치 이미지를 사용하여 원래의 부트 환경으로 부트해야 합니다. [예 9-3](#)을 참조하십시오.
- x86 기반 시스템, **Solaris 10 1/06 릴리스부터** GRUB 메뉴를 사용하는 경우
 - 활성화가 실패하며 GRUB 메뉴는 올바르게 표시되지만 새 부트 환경이 부트 가능하지 않습니다. [예 9-4](#)를 참조하십시오.
 - 활성화가 실패하며 GRUB 메뉴가 표시되지 않습니다. [예 9-5](#)를 참조하십시오.

예 9-1 SPARC: 성공적인 부트 환경 작성 시의 폴백

이 예에서 원래의 `c0t4d0s0` 부트 환경은 성공적으로 활성화되었음에도 불구하고 활성 부트 환경으로 복귀됩니다. 장치 이름은 `first_disk`입니다.

```
# /sbin/luactivate first_disk
# init 6
```

예 9-2 SPARC: 실패한 부트 환경 활성화에서의 폴백

이 예에서는 새 부트 환경을 부트할 수 없는 상태입니다. 원래의 부트 환경인 `c0t4d0s0`에서 부트하기 전에 단일 사용자 모드에서 OK 프롬프트로 복귀해야 합니다.

```
OK boot net -s
# /sbin/luactivate first_disk
Do you want to fallback to activate boot environment c0t4d0s0
(yes or no)? yes
# init 6
```

원래 부트 환경인 `c0t4d0s0`이 활성 부트 환경이 됩니다.

예 9-3 SPARC: DVD, CD 또는 네트워크 설치 이미지를 사용하여 원래 부트 환경으로 폴백

이 예에서는 새 부트 환경을 부트할 수 없는 상태입니다. 원래의 부트 환경에서 부트할 수 없으며 매체나 네트워크 설치 이미지를 사용해야 합니다. 장치는 `/dev/dsk/c0t4d0s0`입니다. 원래 부트 환경인 `c0t4d0s0`이 활성 부트 환경이 됩니다.

예 9-3 SPARC: DVD, CD 또는 네트워크 설치 이미지를 사용하여 원래 부트 환경으로 폴백 (계속)

```
OK boot net -s
# fsck /dev/dsk/c0t4d0s0
# mount /dev/dsk/c0t4d0s0 /mnt
# /mnt/sbin/luactivate
Do you want to fallback to activate boot environment c0t4d0s0
(yes or no)? yes
# umount /mnt
# init 6
```

예 9-4 x86: GRUB 메뉴를 사용하여 원래 부트 환경으로 폴백

Solaris 10 1/06 릴리스부터 다음 예는 GRUB 메뉴를 사용하여 폴백하는 단계를 제공합니다.

이 예에서 GRUB 메뉴는 올바르게 표시되지만 새 부트 환경이 부트 가능하지 않습니다. 폴백을 사용하기 위해 원래 부트 환경이 단일 사용자 모드로 부트됩니다.

1. 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
2. GRUB 메뉴를 표시하려면 시스템을 재부트합니다.

```
# init 6
```

GRUB 메뉴가 표시됩니다.

```
GNU GRUB version 0.95 (616K lower / 4127168K upper memory)
+-----+
|Solaris                                     |
|Solaris failsafe                           |
|second_disk                                 |
|second_disk failsafe                       |
+-----+
Use the ^ and v keys to select which entry is highlighted. Press
enter to boot the selected OS, 'e' to edit the commands before
booting, or 'c' for a command-line.
```

3. GRUB 메뉴에서 원래 부트 환경을 선택합니다. GRUB 소프트웨어를 사용하여 이미 부트 환경이 만들어져 있어야 합니다. **Solaris 10 1/06 릴리스** 이전에 작성된 부트 환경은 GRUB 부트 환경이 아닙니다. 부트 가능한 GRUB 부트 환경이 없는 경우 예 9-5로 넘어갑니다.
4. 다음을 입력하여 GRUB 메뉴를 편집합니다. **e**.
5. 화살표 키를 사용한 다음 **e**를 입력하여 **kernel/boot/multiboot**를 선택합니다. GRUB 편집 메뉴가 표시됩니다.

```
grub edit>kernel /boot/multiboot
```

예 9-4 x86: GRUB 메뉴를 사용하여 원래 부트 환경으로 폴백 (계속)

6. **-s**를 입력하여 단일 사용자 모드로 부트합니다.

```
grub edit>kernel /boot/multiboot -s
```

7. 부트 환경을 부트하고 마운트한 다음 활성화합니다.

```
# b
# fsck /dev/dsk/c0t4d0s0
# mount /dev/dsk/c0t4d0s0 /mnt
# /mnt/sbin/luactivate
Do you want to fallback to activate boot environment c0t4d0s0
(yes or no)? yes
# umount /mnt
# init 6
```

예 9-5 x86: DVD 또는 CD를 사용하여 GRUB 메뉴를 통해 원래 부트 환경으로 폴백

Solaris 10 1/06 릴리스부터 다음 예는 DVD 또는 CD를 사용하여 폴백하는 단계를 제공합니다.

이 예에서는 새 부트 환경을 부트할 수 없는 상태입니다. 또한 GRUB 메뉴가 표시되지 않습니다. 폴백을 사용하기 위해 원래 부트 환경이 단일 사용자 모드로 부트됩니다.

1. Solaris Operating System for x86 Platforms DVD 또는 Solaris Software for x86 Platforms - 1 CD를 넣습니다.
2. 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
3. DVD 또는 CD에서 부트합니다.

```
# init 6
```

GRUB 메뉴가 표시됩니다.

```
GNU GRUB version 0.95 (616K lower / 4127168K upper memory)
+-----+
|Solaris 10 5/09                                     |
|Solaris 10 5/09 Serial Console ttya                 |
|Solaris 10 5/09 Serial Console ttyb (for lx50, v60x and v65x) |
+-----+
Use the ^ and v keys to select which entry is highlighted. Press
enter to boot the selected OS, 'e' to edit the commands before
booting, or 'c' for a command-line.
```

4. 기본 옵션으로 부트할 때까지 기다리거나 표시된 옵션을 모두 선택합니다. 설치 화면이 표시됩니다.

예 9-5 x86: DVD 또는 CD를 사용하여 GRUB 메뉴를 통해 원래 부트 환경으로 폴백 (계속)

```
+-----+
|Select the type of installation you want to perform:      |
|                                                           |
|     1 Solaris Interactive                                |
|     2 Custom JumpStart                                  |
|     3 Solaris Interactive Text (Desktop session)        |
|     4 Solaris Interactive Text (Console session)        |
|     5 Apply driver updates                              |
|     6 Single user shell                                 |
|                                                           |
|     Enter the number of your choice followed by the <ENTER> key. |
|     Alternatively, enter custom boot arguments directly.  |
|                                                           |
|     If you wait 30 seconds without typing anything,      |
|     an interactive installation will be started.         |
+-----+
```

5. “Single user shell” 옵션을 선택합니다.

다음 메시지가 표시됩니다.

```
Do you wish to automatically update the boot archive? y /n
```

6. 다음을 입력합니다. n

```
Starting shell...
#
```

사용자는 이제 단일 사용자 모드에 있습니다.

7. 부트 환경을 마운트합니다. 활성화하고 재부트합니다.

```
# fsck /dev/dsk/c0t4d0s0
# mount /dev/dsk/c0t4d0s0 /mnt
# /mnt/sbin/luactivate
Do you want to fallback to activate boot environment c0t4d0s0
(yes or no)? yes
# umount /mnt
# init 6
```

RAID-1 볼륨의 한쪽 분리 및 업그레이드 예(미러)

이 예에서는 다음 작업을 수행하는 방법에 대해 보여 줍니다.

- 새 부트 환경에 RAID-1 볼륨(미러)을 만듭니다.
- 미러를 차단하고 미러의 한 쪽을 업그레이드합니다.
- 미러(연결)의 다른 한 쪽을 새 미러에 연결합니다.

그림 9-1에서는 세 개의 물리적 디스크가 있는 현재 부트 환경을 보여 줍니다.

RAID-1 볼륨(미러)의 한 쪽을 분리 및 업그레이드

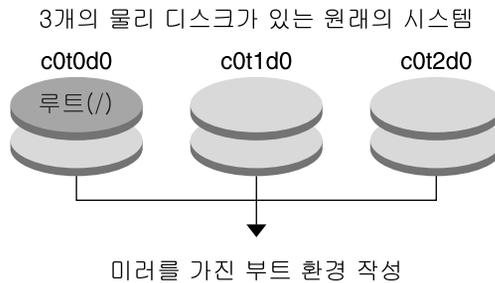


그림 9-1 RAID-1 볼륨(미러)의 한 쪽을 분리 및 업그레이드

1. 미러를 포함하는 새 부트 환경 `second_disk`를 만듭니다.

다음 명령을 사용하여 이러한 작업을 수행합니다.

- `lucreate`는 마운트 포인트 루트(/)에 대해 UFS 파일 시스템을 구성합니다. 미러 `d10`이 생성됩니다. 이 미러는 미러 `d10`에 복사된 현재 부트 환경의 루트(/) 파일 시스템에 대한 저장소입니다. 미러 `d10`에 있는 모든 데이터를 덮어씁니다.
- 두 슬라이스 `c0t0d0s0`과 `c0t1d0s0`은 하위미러로 사용되도록 지정됩니다. 이 두 개의 하위미러는 미러 `d10`에 연결됩니다.

```
# lucreate -c first_disk -n second_disk \
-m /:/dev/md/dsk/d10:ufs,mirror \
-m /:/dev/dsk/c0t1d0s0:attach \
-m /:/dev/dsk/c0t2d0s0:attach
```

2. `second_disk` 부트 환경을 활성화합니다.

```
# /sbin/luactivate second_disk
# init 6
```

3. 다른 부트 환경 `third_disk`를 만듭니다.

다음 명령을 사용하여 이러한 작업을 수행합니다.

- lucreate는 마운트 포인트 루트(/)에 대해 UFS 파일 시스템을 구성합니다. 미러 d20이 생성됩니다.
- c0t1d0s0 슬라이스가 현재 미러에서 제거되고 미러 d20에 추가됩니다. 하위미러의 내용인 루트(/) 파일 시스템이 보존되며 복사는 수행되지 않습니다.

```
# lucreate -n third_disk \  
-m /:/dev/md/dsk/d20:ufs,mirror \  
-m /:/dev/dsk/c0t1d0s0:detach,attach,preserve
```

4. 새 부트 환경 third_disk를 업그레이드합니다.

```
# luupgrade -u -n third_disk \  
-s /net/installmachine/export/Solaris_10/OS_image
```

5. 업그레이드된 부트 환경에 패치를 추가합니다.

```
# luupgrade -t n third_disk -s /net/patches 222222-01
```

6. third_disk 부트 환경을 활성화하여 이 부트 환경을 현재 실행 중인 시스템으로 만듭니다.

```
# /sbin/luactivate third_disk  
# init 6
```

7. second_disk 부트 환경을 삭제합니다.

```
# ludelete second_disk
```

8. 다음 명령을 사용하여 이러한 작업을 수행합니다.

- 미러 d10을 지웁니다.
- c0t2d0s0의 연결 번호를 확인합니다.
- metastat 명령으로 찾은 연결을 미러 d20에 추가합니다. metattach 명령은 새로 추가된 연결과 미러 d20의 연결을 동기화합니다. 연결의 모든 데이터를 덮어씁니다.

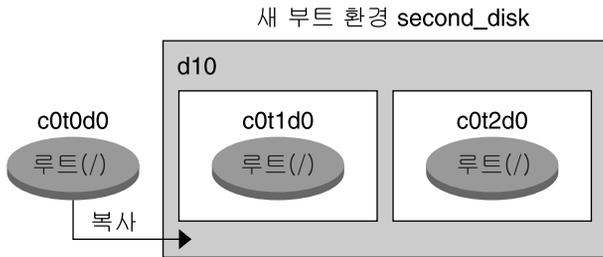
```
# metaclear d10  
# metastat -p | grep c0t2d0s0  
dnum 1 1 c0t2d0s0  
# metattach d20 dnum
```

num metastat 명령에서 찾은 연결 번호입니다.

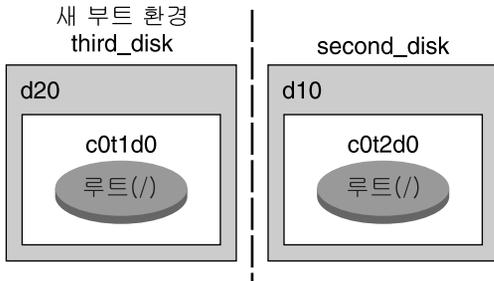
새 부트 환경 third_disk가 업그레이드되었고 현재 실행 중인 시스템입니다. third_disk는 미러된 루트(/) 파일 시스템을 포함합니다.

그림 9-2에서는 앞의 예에 있는 명령을 사용하여 미러를 분리하고 업그레이드하는 전체 프로세스를 보여 줍니다.

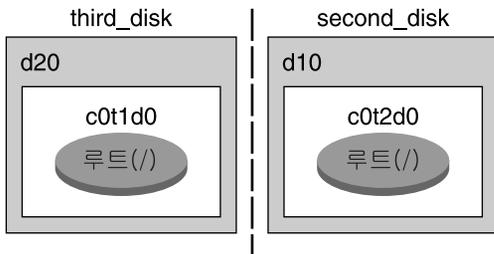
RAID-1 볼륨(미러)의 한 쪽을 분리 및 업그레이드 하는 예(계속)



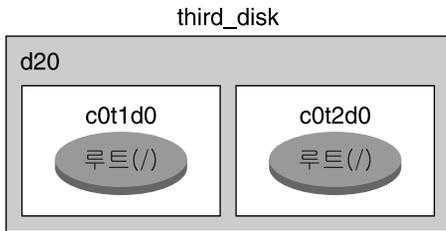
- 새 부트 환경 작성
 - 미러 d10 작성
 - 2개의 단일 슬라이스 연결 작성
 - second_disk 부트 환경 활성화
- 명령:
- ```
lucreate -c first_disk -n second_disk \
-m /:/dev/md/dsk/d10:ufs,mirror \
-m /:/dev/dsk/c0t1d0s0:attach \
-m /:/dev/dsk/c0t2d0s0:attach
/usr/sbin/luactivate second_disk
init 6
```



- 새 부트 환경 third\_disk 작성
  - RAID-0 볼륨 파괴 및 복제 없음
- 명령:
- ```
# lucreate -n third_disk \
-m /:/dev/md/dsk/d20:ufs,mirror \
-m /:/dev/dsk/c0t1d0s0:detach,attach,preserve
```



- third_disk 업그레이드
 - third_disk에 패치 추가
 - third_disk 활성화
- 명령:
- ```
luupgrade -u -n third_disk \
-s /net/installmachine/export/Solaris_10/
luupgrade -t -s /net/patches 222222-01
/usr/sbin/luactivate third_disk
init 6
```



- second\_disk 부트 환경 삭제
  - 연결 c0t2d0 연결
  - c0t2d0를 덮어쓴 미러를 동기화
- 명령:
- ```
# ludelete second_disk
# metaclear d10
# metastat -p | grep c0t2d0s0
# metattach d20 d<num>
```

그림 9-2 RAID-1 볼륨(미러)의 한 쪽을 분리 및 업그레이드(계속)

기존 볼륨에서 Solaris 볼륨 관리자 RAID-1 볼륨으로의 마이그레이션 예

Solaris Live Upgrade를 사용하면 RAID-1 볼륨(미러)에 새 부트 환경을 만들 수 있습니다. 현재 부트 환경의 파일 시스템은 다음 중 하나일 수 있습니다.

- 물리적 저장 장치
- Solaris 볼륨 관리자 제어 RAID-1 볼륨
- Veritas VXFS 제어 볼륨

하지만 새 부트 환경 대상은 Solaris 볼륨 관리자 RAID-1 볼륨이어야 합니다. 예를 들어, 루트(/) 파일 시스템의 복사본에 지정된 슬라이스는 /dev/vx/dsk/rootvol이어야 합니다. rootvol은 루트(/) 파일 시스템을 포함하는 볼륨입니다.

이 예에서 현재 부트 환경은 Solaris 볼륨 관리자 볼륨이 아닌 루트(/) 파일 시스템을 포함합니다. 새 부트 환경은 Solaris Volume Manager RAID-1 볼륨 c0t2d0s0에서 루트(/) 파일 시스템을 포함하여 만들어집니다. lucreate 명령은 현재 볼륨을 Solaris 볼륨 관리자 볼륨으로 이전합니다. 새 부트 환경의 이름은 svm_be입니다. lustatus 명령은 새 부트 환경이 활성화 및 재부트 준비가 되었는지 여부를 보고합니다. 새 부트 환경이 활성화되어 현재 부트 환경이 됩니다.

```
# lucreate -n svm_be -m /:/dev/md/dsk/d1:mirror,ufs \
-m /:/dev/dsk/c0t2d0s0:attach
# lustatus
# luactivate svm_be
# lustatus
# init 6
```

빈 부트 환경 만들기 및 Solaris Flash 아카이브 설치 예

다음은 세 단계로 구성된 절차입니다.

- 빈 부트 환경 만들기
- 아카이브 설치
- 부트 환경을 활성화하여 현재 실행 부트 환경으로 만들기

lucreate 명령을 활성 부트 환경의 파일 시스템을 기준으로 부트 환경을 작성합니다. lucreate 명령을 -s 옵션과 함께 사용하면 lucreate는 빈 부트 환경을 빨리 만듭니다. 해당 슬라이스는 지정된 파일 시스템에 대해 예약되지만 파일 시스템은 복사되지 않습니다. 부트 환경의 이름은 지정되지만 Solaris Flash 아카이브를 사용하여 설치될 때까지 실제로 만들어지지 않습니다. 아카이브를 사용하여 빈 부트 환경이 설치되면 파일 시스템은 예약된 슬라이스에 설치됩니다. 그런 다음 부트 환경이 활성화됩니다.

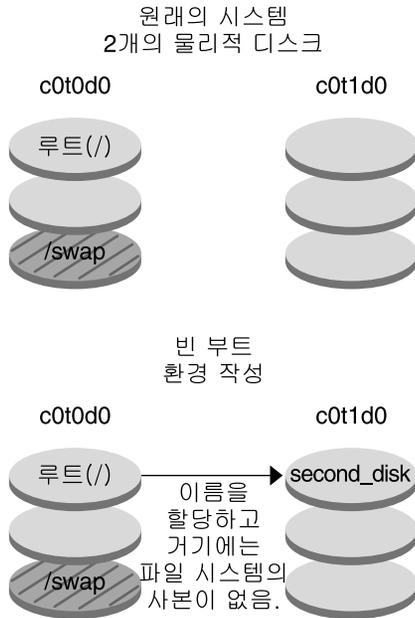
빈 부트 환경 만들기

첫 단계에서 빈 부트 환경이 만들어집니다. 슬라이스는 지정된 파일 시스템용으로 예비되지만 현재 부트 환경에서 복사되는 파일 시스템은 없습니다. 새 부트 환경 이름은 `second_disk`로 지정됩니다.

```
# lucreate -s - -m /:/dev/dsk/c0t1d0s0:ufs \
-n second_disk
```

이제 부트 환경을 Solaris Flash 아카이브로 채울 수 있습니다.

그림 9-3에서는 빈 부트 환경 만들기를 보여 줍니다.



```
Command: # lucreate
          -m /:/dev/dsk/c0t1d0s0:ufs \
          -n second_disk
```

그림 9-3 빈 부트 환경 만들기

새 부트 환경에 Solaris Flash 아카이브 설치

두 번째 단계에서 앞의 예에서 만든 `second_disk` 부트 환경에 아카이브를 설치합니다. 아카이브는 로컬 시스템에 있습니다. `-s` 및 `-a` 옵션용 운영 체제 버전은 모두 Solaris 10 5/09 릴리스입니다. 아카이브의 이름은 `Solaris_10.flar`로 지정됩니다.

```
# luupgrade -f -n second_disk \
-s /net/installmachine/export/Solaris_10/OS_image \
-a /net/server/archive/10.flar
```

부트 환경이 활성화될 준비가 되었습니다.

새 부트 환경 활성화

마지막 단계에서 `second_disk` 부트 환경은 `luactivate` 명령으로 부트할 수 있습니다. 그런 다음 시스템이 재부트되고 `second_disk`가 아카이브 부트 환경이 됩니다.

```
# luactivate second_disk
# init 6
```

- 빈 부트 환경 만들기에 대한 단계별 정보는 75 페이지 “Solaris Flash 아카이브용 빈 부트 환경 만들기”를 참조하십시오.
- Solaris Flash 아카이브 만들기에 대한 단계별 설명은 **Solaris 10 5/09 설치 설명서: Solaris Flash 아카이브(작성 및 설치)**의 3 장, “Solaris Flash 아카이브 만들기(작업)”를 참조하십시오.
- 부트 환경 활성화 또는 원래 부트 환경으로 폴백에 대한 단계별 설명은 6 장, “오류 복구: 원래 부트 환경으로 폴백(작업)”을 참조하십시오.

Solaris Live Upgrade(명령 참조)

다음 목록에는 명령줄에 입력할 수 있는 명령이 표시됩니다. Solaris Live Upgrade에는 목록의 모든 명령줄 유틸리티에 대한 매뉴얼 페이지가 있습니다.

Solaris Live Upgrade 명령줄 옵션

작업	명령
비활성 부트 환경을 활성화합니다.	<code>luactivate(1M)</code>
예약된 복사를 취소하거나 작업을 만듭니다.	<code>lucancel(1M)</code>
활성 부트 환경을 비활성 부트 환경과 비교합니다.	<code>lucompare(1M)</code>
파일 시스템을 다시 복사하여 비활성 부트 환경을 업데이트합니다.	<code>lumake(1M)</code>
부트 환경을 만듭니다.	<code>lucreate(1M)</code>
비활성 부트 환경의 이름을 지정합니다.	<code>lucurr(1M)</code>
부트 환경을 삭제합니다.	<code>ludelete(1M)</code>
부트 환경 이름에 설명을 추가합니다.	<code>ludesc(1M)</code>
각 부트 환경에 대한 핵심 파일 시스템을 나열합니다.	<code>lufslist(1M)</code>
부트 환경의 모든 파일 시스템을 마운트할 수 있도록 합니다. 이 명령을 사용하여 부트 환경이 비활성 상태에서 부트 환경의 파일을 수정할 수 있습니다.	<code>lumount(1M)</code>
부트 환경의 이름을 바꿉니다.	<code>lurename(1M)</code>

작업	명령
모든 부트 환경의 상태를 나열합니다.	<code>lustatus(1M)</code>
부트 환경의 모든 파일 시스템에서 마운트를 해제할 수 있습니다. 이 명령을 사용하여 부트 환경이 비활성 상태에서 부트 환경의 파일을 수정할 수 있습니다.	<code>luumount(1M)</code>
비활성 부트 환경에서 OS를 업그레이드하거나 Flash 아카이브를 설치할 수 있습니다.	<code>luupgrade(1M)</code>

2

Solaris Live Upgrade를 사용하여 업그레이드 및 ZFS 루트 풀로 마이그레이션

여기에서는 Solaris Live Upgrade를 사용하여 ZFS 저장소 풀에서 비활성 부트 환경을 만들고 업그레이드하는 데 대한 개요와 지침을 제공합니다. 또한 UFS 루트(/) 파일 시스템을 ZFS 루트 풀로 마이그레이션할 수도 있습니다.

Solaris Live Upgrade 및 ZFS(개요)

Solaris Live Upgrade를 사용하여 UFS 파일 시스템을 ZFS 루트 풀로 마이그레이션하고 기존 ZFS 루트 풀에서 ZFS 루트 파일 시스템을 만들 수 있습니다.

주 - Solaris Live Upgrade를 사용하여 부트 환경을 만드는 작업은 **Solaris 10 10/08 릴리스**의 새로운 기능입니다. UFS 파일 시스템에서 Solaris Live Upgrade를 수행하면 명령줄 매개 변수와 Solaris Live Upgrade의 작동이 둘 다 변경되지 않는 상태로 유지됩니다. Solaris Live Upgrade를 UFS 파일 시스템이 있는 시스템에서 수행하려면 이 설명서의 제1부를 참조하십시오.

다음 절에서는 이러한 작업에 대한 개요를 제공합니다.

- 174 페이지 “ZFS와 함께 Solaris Live Upgrade 사용 소개”.
- SVM 볼륨의 사용 여부와 관계 없이 UFS 파일 시스템을 마이그레이션할 수 있습니다.
 - 174 페이지 “UFS 파일 시스템에서 ZFS 루트 풀로 마이그레이션”
 - 176 페이지 “Solaris Volume Manager 볼륨이 구성된 UFS 파일 시스템을 ZFS 루트 파일 시스템으로 마이그레이션”
- 기존 루트 풀이나 다른 루트 풀에 새 ZFS 부트 환경을 만들 수 있습니다.
 - 177 페이지 “동일한 루트 풀 내에 새 부트 환경 만들기”
 - 179 페이지 “다른 루트 풀에 새 부트 환경 만들기”
- 181 페이지 “현재 실행 중인 시스템 이외의 소스에서 새 부트 환경 만들기”.
- 182 페이지 “비전역 영역이 설치된 시스템에서 ZFS 부트 환경 만들기”.

ZFS와 함께 Solaris Live Upgrade 사용 소개

UFS 파일 시스템이 있는 경우 Solaris Live Upgrade는 이전 릴리스와 동일하게 작동합니다. 이제 UFS 파일 시스템에서 ZFS 루트 풀로 마이그레이션하고 ZFS 루트 풀에 새 부트 환경을 만들 수 있습니다. 이러한 작업을 위해 `lucreate` 명령이 `-p` 옵션을 사용하여 향상되었습니다. 명령 구문은 다음과 같습니다.

```
# lucreate [-c active_BE_name] -n BE_name [-p zfs_root_pool]
```

`-p` 옵션은 새 부트 환경이 있는 ZFS 풀을 지정합니다. 소스와 대상 부트 환경이 동일한 풀에 있는 경우 이 옵션은 생략할 수 있습니다.

`lucreate` 명령의 `-m` 옵션은 ZFS에서 지원되지 않습니다. 기타 `lucreate` 명령 옵션은 일부를 제외하고는 보통 때처럼 작동합니다. 제한 사항에 대한 자세한 내용은 [183 페이지 “Solaris Live Upgrade 사용 시 시스템 요구 사항 및 제한”](#)를 참조하십시오.

UFS 파일 시스템에서 ZFS 루트 풀로 마이그레이션

현재 실행 중인 시스템에서 부트 환경을 만드는 경우 `lucreate` 명령은 UFS 루트(/) 파일 시스템을 ZFS 루트 풀로 복사합니다. 복사 프로세스는 사용 중인 시스템에 따라 시간이 걸릴 수 있습니다.

UFS 파일 시스템에서 마이그레이션하는 경우 소스 부트 환경은 디스크 슬라이스에서 UFS 루트(/) 파일 시스템일 수 있습니다. ZFS 루트 풀의 소스 부트 환경에서 UFS 파일 시스템의 부트 환경을 만들 수 없습니다.

UFS 루트(/) 파일 시스템에서 ZFS 루트 풀로 마이그레이션

다음 명령은 ZFS 루트 풀을 만들고, 해당 ZFS 루트 풀의 UFS 루트(/) 파일 시스템에서 새 부트 환경을 만듭니다. ZFS 루트 풀은 `lucreate` 작업 전에 있어야 하며, 업그레이드 가능하고 부트 가능하도록 하려면 전체 디스크가 아닌 슬라이스로 만들어야 합니다. 디스크에는 EFI 레이블이 있을 수 없지만 SMI 레이블이 있어야 합니다. 제한 사항에 대한 자세한 내용은 [183 페이지 “Solaris Live Upgrade 사용 시 시스템 요구 사항 및 제한”](#)를 참조하십시오.

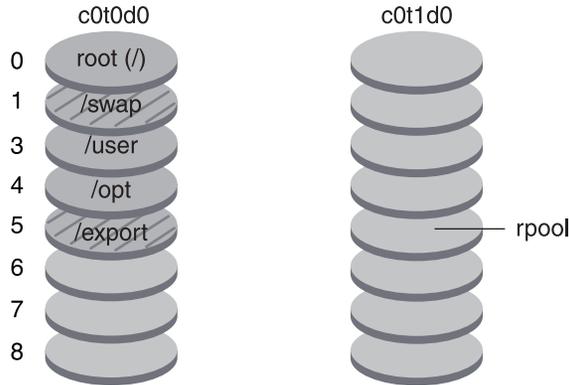
[그림 11-1](#)에서는 별도의 슬라이스 `c0t1d0s5`에 루트 풀 `rpool`을 만드는 `zpool` 명령을 보여 줍니다. 디스크 슬라이스 `c0t0d0s0`에는 UFS 루트(/) 파일 시스템이 들어 있습니다.

`lucreate` 명령에서 `-c` 옵션은 현재 실행 중인 시스템 `c0t0d0`, 즉 UFS 루트(/) 파일 시스템의 이름을 지정합니다. `-n` 옵션은 만들려는 부트 환경에 이름 `new-zfsBE`를 할당합니다. `-p` 옵션은 새 부트 환경 `rpool`을 배치할 위치를 지정합니다. UFS/export 파일 시스템과 /swap 볼륨은 새 부트 환경에 복사되지 않습니다.

UFS 파일 시스템에서 ZFS 루트 풀로 마이그레이션

① 슬라이스에 ZFS 루트 풀을 만듭니다.

```
# zpool create rpool c0t1d0s5
```



② UFS 부트 환경을 rpool로 복사합니다.

```
# lucreate -c c0t0d0 -n new-zfsBE -p rpool
```

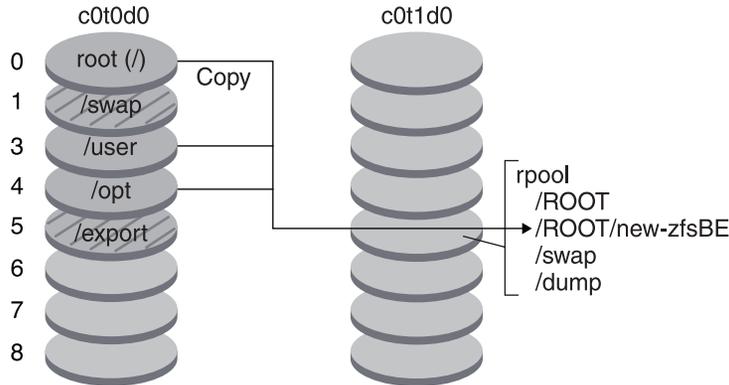


그림 11-1 UFS 파일 시스템에서 ZFS 루트 풀로 마이그레이션

예 11-1 UFS 루트(/) 파일 시스템에서 ZFS 루트 풀로 마이그레이션

이 예에서는 **그림 11-1**에서와 동일한 명령을 보여 줍니다. 명령은 새 루트 풀 rpool을 만들고 UFS 루트(/) 파일 시스템의 풀에 새 부트 환경을 만듭니다. 이 예에서 `zfs list` 명령은 `zpool` 명령을 통해 만든 ZFS 루트 풀을 보여 줍니다. 다음 `zfs list` 명령은 `lucreate` 명령을 통해 만든 데이터 세트를 보여 줍니다.

```
# zpool create rpool c0t1d0s5
# zfs list
```

예 11-1 UFS 루트(/) 파일 시스템에서 ZFS 루트 풀로 마이그레이션 (계속)

```

NAME                USED  AVAIL  REFER  MOUNTPOINT
rpool                9.29G 57.6G   20K    /rpool

# lucreate -c c0t0d0 -n new-zfsBE -p rpool
# zfs list
NAME                USED  AVAIL  REFER  MOUNTPOINT
rpool                9.29G 57.6G   20K    /rpool
rpool/ROOT           5.38G 57.6G   18K    /rpool/ROOT
rpool/ROOT/new-zfsBE 5.38G 57.6G  551M   /tmp/.alt.luupdall.110034
rpool/dump           1.95G   -    1.95G   -
rpool/swap           1.95G   -    1.95G   -

```

새 부트 환경은 rpool/ROOT/new-zfsBE입니다. 부트 환경 new-zfsBE는 이제 업그레이드하고 활성화할 수 있습니다.

Solaris Volume Manager 볼륨이 구성된 UFS 파일 시스템을 ZFS 루트 파일 시스템으로 마이그레이션

시스템에 SVM(Solaris Volume Manager)의 볼륨이 있는 경우 UFS 파일 시스템을 마이그레이션할 수 있습니다. 기존 SVM 구성에서 UFS 부트 환경을 만들려면 현재 실행 중인 시스템에서 새 부트 환경을 만듭니다. 그런 다음 새 UFS 부트 환경에서 ZFS 부트 환경을 만듭니다.

SVM(Solaris Volume Manager)의 개요. ZFS에서는 물리적 저장소를 관리하기 위해 저장소 풀 개념을 사용합니다. 기존의 파일 시스템은 단일 물리적 장치의 맨 위에서 구성되었습니다. 여러 장치의 주소를 지정하고 데이터 중복성을 제공하는 볼륨 관리자의 개념이 도입되어 단일 장치의 이미지가 제공됩니다. 따라서 여러 장치를 사용하기 위해 파일 시스템을 수정하지 않아도 되지만 이 설계로 인해 계층이 복잡해졌습니다. 해당 파일 시스템에 가상 볼륨에 있는 데이터의 물리적 배치에 대한 제어 권한이 없기에 이 복잡성으로 인해 특정 파일 시스템이 결국 향상되지 못했습니다.

SVM을 대체하는 ZFS 저장소 풀. ZFS는 볼륨 관리를 완벽하게 제거했습니다. 사용자에게 가상 볼륨을 만들도록 강제하는 대신 ZFS는 장치를 저장소 풀에 결합합니다. 저장소 풀은 저장소 장치 레이어 및 데이터 중복성과 같은 물리적 특성에 대해 설명하고, 파일 시스템을 만들 수 있는 임의의 데이터 저장소 역할을 수행합니다. 파일 시스템은 개별 장치에 대해 더 이상 제약을 받지 않으며, 장치를 활성화하여 풀에서 모든 파일 시스템과 공간을 공유합니다. 파일 시스템은 저장소 풀에 할당된 공간 내에서 자동으로 커치프로 파일 시스템의 크기를 미리 결정하지 않아도 됩니다. 새 저장소가 추가되면 풀에 있는 모든 파일 시스템은 추가 작업 없이 즉시 추가 공간을 사용할 수 있습니다. 저장소 풀은 여러 가지 방법으로 가상 메모리 시스템 역할을 합니다. 메모리 DIMM이 시스템에 추가되면 운영 체제는 사용자가 몇 가지 명령을 호출하여 메모리를 구성하고 개별 프로세스에 메모리를 할당하도록 강제하지 않습니다. 모든 시스템 프로세스에서 추가 메모리가 자동으로 사용됩니다.

예 11-2 SVM 볼륨이 있는 UFS 루트(/) 파일 시스템에서 ZFS 루트 풀로 마이그레이션

SVM 볼륨이 있는 시스템을 마이그레이션하는 경우 SVM 볼륨은 무시됩니다. 다음 예에서와 같이 루트 풀 내에서 미러를 설정할 수 있습니다.

이 예에서 `lucreate` 명령은 `-m` 옵션과 같이 사용되어 현재 실행 중인 시스템에서 새 부트 환경을 만듭니다. 디스크 슬라이스 `c1t0d0s0`에는 SVM 볼륨으로 구성된 UFS 루트(/) 파일 시스템이 들어 있습니다. `zpool` 명령은 루트 풀 `c1t0d0s0`과 RAID-1 볼륨(미러) `c2t0d0s0`을 만듭니다. 두 번째 `lucreate` 명령에서 `-n` 옵션은 만들려는 부트 환경에 이름 `c0t0d0s0`을 지정합니다. `-s` 옵션은 UFS 루트(/) 파일 시스템을 식별합니다. `-p` 옵션은 새 부트 환경 `rpool`을 배치할 위치를 지정합니다.

```
# lucreate -n ufsBE -m /:/dev/md/dsk/d104:ufs
# zpool create rpool mirror c1t0d0s0 c2t0d0s0
# lucreate -n c0t0d0s0 -s ufsBE -p rpool
```

부트 환경 `c0t0d0s0`은 이제 업그레이드하고 활성화할 수 있습니다.

ZFS 루트 풀에서 새 부트 환경 만들기

동일한 루트 풀이나 새 루트 풀에 새 ZFS 부트 환경을 만들 수 있습니다. 이 절에서는 다음에 대한 개요를 제공합니다.

- 177 페이지 “동일한 루트 풀 내에 새 부트 환경 만들기”
- 179 페이지 “다른 루트 풀에 새 부트 환경 만들기”

동일한 루트 풀 내에 새 부트 환경 만들기

동일한 ZFS 루트 풀에서 새 부트 환경을 만드는 경우 `lucreate` 명령은 소스 부트 환경에서 스냅샷을 만들고, 복제는 스냅샷에서 만들어집니다. 스냅샷과 복제는 거의 즉시 만들어지며, 디스크 공간은 최소로 사용됩니다. 결국 필요한 디스크 공간의 크기는 업그레이드 프로세스의 일부로 교체되는 파일 수에 따라 달라집니다. 스냅샷은 읽기 전용이지만 복제는 스냅샷의 읽기/쓰기 복사본입니다. 복제 부트 환경에서 변경된 내용은 스냅샷이나 스냅샷이 만들어진 소스 부트 환경에 반영되지 않습니다.

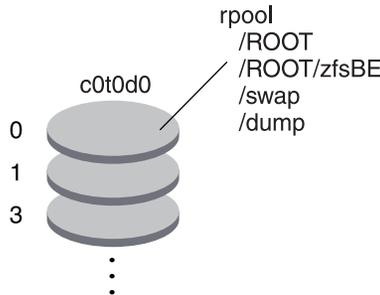
주 - 활성화 데이터 세트에서 데이터가 변경되면 스냅샷은 기존 데이터를 계속 참조하여 공간을 소모합니다. 결과적으로 스냅샷으로 인해 해당 데이터가 풀에 다시 해제되지 않습니다. 스냅샷에 대한 자세한 내용은 [Solaris ZFS Administration Guide](#)의 7 장, “Working With ZFS Snapshots and Clones”를 참조하십시오.

현재 부트 환경이 동일한 ZFS 풀에 있는 경우 `-p` 옵션을 생략합니다.

그림 11-2에서는 ZFS 루트 풀에서 ZFS 부트 환경 만들기를 보여 줍니다. 슬라이스 c0t0d0s0에는 ZFS 루트 풀 rpool이 들어 있습니다. lucreate 명령에서 -n 옵션은 만들려는 부트 환경에 이름 new-zfsBE를 할당합니다. 원래 루트 풀의 스냅샷은 rpool@new-zfsBE로 만들어집니다. 스냅샷은 새 부트 환경 new-zfsBE인 복제를 만드는 데 사용됩니다. 부트 환경 new-zfsBE는 이제 업그레이드하고 활성화할 수 있습니다.

동일한 루트 풀에 새 부트 환경 만들기

현재 실행 중인 시스템



① rpool에 새 부트 환경을 만듭니다.

```
# lucreate -n new-zfsBE
```

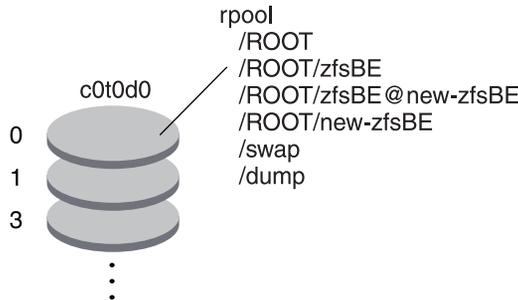


그림 11-2 동일한 루트 풀에서 새 부트 환경 만들기

예 11-3 동일한 ZFS 루트 풀 내에 부트 환경 만들기

이 예에서는 동일한 루트 풀에서 새 부트 환경을 만드는 그림 11-2에서와 동일한 명령을 보여 줍니다. lucreate 명령은 -c zfsBE 옵션을 사용하여 현재 실행 중인 부트 환경의 이름을 지정하며, -n new-zfsBE는 새 부트 환경을 만듭니다. zfs list 명령은 새 부트 환경과 스냅샷이 있는 ZFS 데이터 세트를 보여 줍니다.

```
# lucreate -c zfsBE -n new-zfsBE
```

```
# zfs list
```

예 11-3 동일한 ZFS 루트 풀 내에 부트 환경 만들기 (계속)

AME	USED	AVAIL	REFER	MOUNTPOINT
rpool	9.29G	57.6G	20K	/rpool
rpool/ROOT	5.38G	57.6G	18K	/rpool/ROOT
rpool/ROOT/zfsBE	5.38G	57.6G	551M	
rpool/ROOT/zfsBE@new-zfsBE	66.5K	-	551M	-
rpool/ROOT/new-zfsBE	5.38G	57.6G	551M	/tmp/.alt.luupdall.110034
rpool/dump	1.95G	-	1.95G	-
rpool/swap	1.95G	-	1.95G	-

다른 루트 풀에 새 부트 환경 만들기

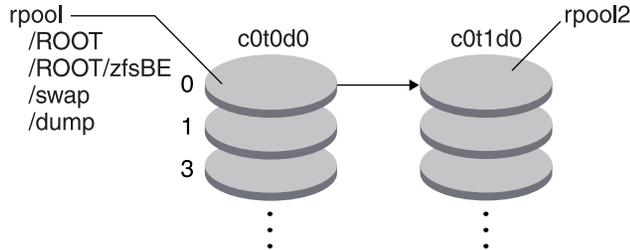
lucreate 명령을 사용하여 기존 ZFS 루트 풀을 다른 ZFS 루트 풀로 복사할 수 있습니다. 복사 프로세스는 사용 중인 시스템에 따라 약간 시간이 걸릴 수 있습니다.

그림 11-3에서는 부트 가능한 ZFS 루트 풀이 아직 없기 때문에 `c0t1d0s5`에 ZFS 루트 풀 `rpool2`를 만드는 `zpool` 명령을 보여 줍니다. `lucreate` 명령을 `-n` 옵션과 함께 사용하면 만들려는 부트 환경에 `new-zfsBE` 이름을 할당합니다. `-p` 옵션은 새 부트 환경을 배치할 위치를 지정합니다.

다른 루트 풀에 새 부트 환경 만들기

- ① 별도의 슬라이스에 새 루트 풀을 만듭니다.

```
# zpool create rpool2 c0t1d0s0
```



- ② Rpool2에 새 부트 환경을 만듭니다.

```
# lucreate -n new-zfsBE -p rpool2
```

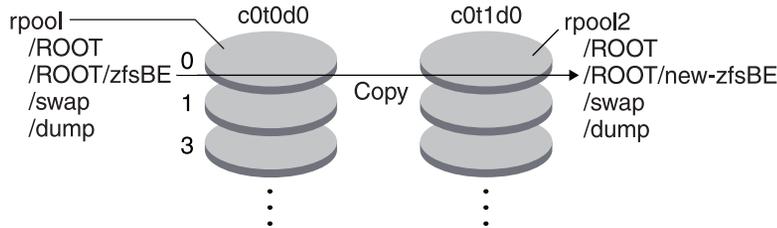


그림 11-3 다른 루트 풀에 새 부트 환경 만들기

예 11-4 서로 다른 ZFS 루트 풀에 부트 환경 만들기

이 예에서는 새 루트 풀을 만들고 새로 만든 루트 풀에서 새 부트 환경을 만드는, [그림 11-3](#)에서와 동일한 명령을 보여 줍니다. 이 예에서 `zpool create` 명령은 `rpool2`를 만듭니다. `zfs list` 명령은 `rpool2`에서 만든 ZFS 데이터 세트가 없음을 보여 줍니다. 데이터 세트는 `lucreate` 명령으로 만들어집니다.

```
# zpool create rpool2 c0t2d0s5
```

```
# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool2	9.29G	57.6G	20K	/rpool2
rpool	9.29G	57.6G	20K	/.new.lulib.rs.109262
rpool/ROOT	5.46G	57.6G	18K	legacy
rpool/ROOT/zfsBE	5.46G	57.6G	551M	
rpool/dump	3.99G	-	3.99G	-
rpool/swap	3.99G	-	3.99G	-

새 ZFS 루트 풀 `rpool2`는 디스크 슬라이스 `c0t2d0s5`에 만들어집니다.

예 11-4 서로 다른 ZFS 루트 풀에 부트 환경 만들기 (계속)

```
# lucreate -n new-zfsBE -p rpool2
# zfs list
NAME                                USED    AVAIL    REFER  MOUNTPOINT
rpool2                               9.29G   57.6G    20K    /rpool2
rpool2/ROOT/                         5.38G   57.6G    18K    /rpool2/ROOT
rpool2/ROOT/new-zfsBE                 5.38G   57.6G    551M   /tmp/.new.luupdall.109859
rpool2/dump                           3.99G   -        3.99G  -
rpool2/swap                           3.99G   -        3.99G  -
rpool                                 9.29G   57.6G    20K    /.new.lulib.rs.109262
rpool/ROOT                           5.46G   57.6G    18K    legacy
rpool/ROOT/zfsBE                      5.46G   57.6G    551M
rpool/dump                            3.99G   -        3.99G  -
rpool/swap                            3.99G   -        3.99G  -
```

새 부트 환경 `new-zfsBE`는 다른 데이터 세트 `ROOT`, `dump` 및 `swap`과 함께 `rpool2`에 만들어집니다. 부트 환경 `new-zfsBE`는 이제 업그레이드하고 활성화할 수 있습니다.

현재 실행 중인 시스템 이외의 소스에서 새 부트 환경 만들기

부트 환경을 현재 실행 중인 시스템 이외의 소스에서 만들려는 경우 `lucreate` 명령을 `-s` 옵션과 함께 사용해야 합니다. `-s` 옵션은 UFS 파일 시스템의 경우와 동일하게 작동합니다. `-s` 옵션은 대체 루트(/) 파일 시스템의 경로를 제공합니다. 이 대체 루트(/) 파일 시스템은 새 ZFS 루트 풀을 만들기 위한 소스입니다. 대체 루트는 UFS(/) 루트 파일 시스템이거나 ZFS 루트 풀일 수 있습니다. 복사 프로세스는 사용 중인 시스템에 따라 시간이 걸릴 수 있습니다.

예 11-5 대체 루트(/) 파일 시스템에서 부트 환경 만들기

다음 명령은 기존 ZFS 루트 풀에서 새 ZFS 루트 풀을 만듭니다. `-n` 옵션은 만들려는 부트 환경에 이름 `new-zfsBE`를 할당합니다. `-s` 옵션은 부트 환경 `source-zfsBE`가 현재 실행 중인 부트 환경 대신 복사본의 소스로 사용되도록 지정합니다. `-p` 옵션은 `newpool2`에 새 부트 환경을 배치하도록 지정합니다.

```
# lucreate -n new-zfsBE -s source-zfsBE -p rpool2
```

부트 환경 `new-zfsBE`는 이제 업그레이드하고 활성화할 수 있습니다.

비전역 영역이 설치된 시스템에서 ZFS 부트 환경 만들기

Solaris Live Upgrade를 사용하여 비전역 영역을 ZFS 루트 파일 시스템으로 마이그레이션할 수 있습니다. 개요, 계획 및 단계별 절차는 14 장, “비전역 영역이 설치된 ZFS용 Solaris Live Upgrade”를 참조하십시오.

추가 자원

이 장에 포함된 항목에 대한 추가 정보는 표 11-1에 나열된 자원을 참조하십시오.

표 11-1 추가 자원

자원	위치
개요, 계획, 단계별 지침을 비롯한 ZFS 정보	Solaris ZFS Administration Guide
UFS 파일 시스템이 있는 시스템에서 Solaris Live Upgrade 사용	이 설명서의 제1부

◆◆◆ 12 장

ZFS용 Solaris Live Upgrade(계획)

이 장에서는 UFS 파일 시스템을 ZFS 파일 시스템으로 마이그레이션하기 전이나 기존 ZFS 루트 풀에서 새 ZFS 부트 환경을 만들기 전에 검토용으로 필요한 지침과 요구 사항에 대해 제공합니다.

주 - Solaris Live Upgrade를 사용하여 부트 환경을 만드는 작업은 **Solaris 10 10/08 릴리스**의 새로운 기능입니다. UFS 파일 시스템에서 Solaris Live Upgrade를 수행하면 명령줄 매개 변수와 Solaris Live Upgrade의 작동이 둘 다 변경되지 않는 상태로 유지됩니다. Solaris Live Upgrade를 UFS 파일 시스템이 있는 시스템에서 수행하려면 이 설명서의 **제1부**를 참조하십시오.

Solaris Live Upgrade 사용 시 시스템 요구 사항 및 제한

UFS 파일 시스템을 ZFS 파일 시스템으로 마이그레이션하기 전이나 기존 ZFS 루트 풀에서 새 ZFS 부트 환경을 만들기 전에 다음 요구 사항과 제한 사항을 읽고 이해해야 합니다. 이러한 요구 사항은 **Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획의 6 장**, “ZFS 루트 파일 시스템 설치(계획)”에 나열된 요구 사항에 대한 추가 사항입니다.

표 12-1 요구 사항 및 제한 사항

요구 사항 또는 제한 사항	설명	정보
최소 Solaris 10 10/08 릴리스가 설치되어 있어야 합니다.	Solaris Live Upgrade를 사용하여 UFS 파일 시스템에서 ZFS 루트 풀로 마이그레이션하거나 루트 풀에서 새 부트 환경을 만드는 작업은 Solaris 10 10/08 릴리스 의 새로운 기능입니다. 이 릴리스에는 ZFS와 함께 Solaris Live Upgrade를 사용하는 데 필요한 소프트웨어가 포함되어 있습니다. ZFS를 사용하려면 최소 이 릴리스가 설치되어 있어야 합니다.	

표 12-1 요구 사항 및 제한 사항 (계속)

요구 사항 또는 제한 사항	설명	정보
디스크 공간	부트 가능 ZFS 루트 파일 시스템에서 사용 가능한 최소 풀 공간 크기는 물리적 메모리의 양, 사용 가능한 디스크 공간, 만들어지는 부트 환경 수에 따라 달라집니다.	자세한 내용은 Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획 의 “ZFS 설치의 디스크 공간 요구 사항”을 참조하십시오.
UFS 루트 (/) 파일 시스템에서 ZFS 루트 풀로 마이그레이션하는 경우 이러한 요구 사항에 대해 고려하십시오.	<ul style="list-style-type: none"> ■ UFS 파일 시스템에서 ZFS 파일 시스템으로의 마이그레이션만 가능합니다. <ul style="list-style-type: none"> ■ UFS 파일 시스템 이외의 파일 시스템은 ZFS 루트 풀로 전송할 수 없습니다. ■ UFS 파일 시스템은 ZFS 루트 풀에서 만들 수 없습니다. ■ 마이그레이션을 수행하기 전에 ZFS 저장소 풀이 있어야 합니다. ■ ZFS 저장소 풀을 업그레이드 가능하고 부트 가능하도록 하려면 전체 디스크가 아닌 슬라이스로 만들어야 합니다. <ul style="list-style-type: none"> ■ 슬라이스로 만들어진 풀은 미러될 수 있지만 RAID-Z 또는 여러 디스크의 비중복 구성의 경우는 가능하지 않습니다. SVM 장치 정보를 /dev/md/[r]dsk 디렉토리에 사용할 수 있어야 합니다. ■ 풀에는 SMI 레이블이 있어야 합니다. EFI-레이블이 있는 디스크는 부트될 수 없습니다. ■ x86에만 해당: ZFS 풀은 fdisk 분할 영역이 있는 슬라이스에 있어야 합니다. 	<ul style="list-style-type: none"> ■ 단계별 절차는 187 페이지 “UFS 파일 시스템을 ZFS 파일 시스템으로 마이그레이션”을 참조하십시오. ■ SVM에 대한 자세한 내용은 Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획의 “Solaris Volume Manager 구성요소의 개요”를 참조하십시오. ■ fdisk 분할 영역에 대한 자세한 내용은 Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획의 “분할 권장 사항”을 참조하십시오. ■ EFI-레이블이 있는 디스크의 기타 제한 사항에 대한 자세한 내용은 System Administration Guide: Devices and File Systems의 “Overview of Disk Management”를 참조하십시오.
공유 파일 시스템을 마이그레이션하는 경우 공유 파일 시스템을 새 ZFS 루트 풀에서 별도의 슬라이스로 복사할 수 없습니다.	예를 들어 UFS 루트(/) 파일 시스템에서 Solaris Live Upgrade를 수행하는 경우 -m 옵션을 사용하여 /export 파일 시스템을 다른 장치로 복사할 수 있습니다. 사용자에게는 공유 파일 시스템을 ZFS 풀로 복사하는 -m 옵션이 없습니다.	

표 12-1 요구 사항 및 제한 사항 (계속)

요구 사항 또는 제한 사항	설명	정보
비전역 영역이 포함된 UFS 루트 파일 시스템을 마이그레이션하는 경우 공유 파일 시스템은 마이그레이션되지 않습니다.	UFS 루트(/) 파일 시스템과 비전역 영역이 설치된 시스템에서 비전역 영역이 UFS-ZFS 마이그레이션의 일부로 중요 파일 시스템에 있는 경우 해당 영역이 마이그레이션됩니다. 그렇지 않고 동일한 ZFS 풀에서 업그레이드하는 경우에는 영역이 복제됩니다. 비전역 영역이 공유 UFS(/) 파일 시스템에 있는 경우 ZFS 루트 풀로 마이그레이션하려면 먼저 이전 Solaris 릴리스에서와 같이 해당 영역을 업그레이드해야 합니다.	<ul style="list-style-type: none"> ■ 중요 파일 시스템과 공유 파일 시스템에 대한 자세한 내용은 22 페이지 “파일 시스템 유형”를 참조하십시오. ■ 비전역 영역이 설치되어 있는 경우 단계별 지침은 14 장, “비전역 영역이 설치된 ZFS용 Solaris Live Upgrade”를 참조하십시오.
ZFS rename 명령을 사용하지 마십시오.	Solaris Live Upgrade 기능은 이름 변경을 인식하지 못하며 <code>ludetele</code> 와 같은 후속 명령이 실패합니다. 실제로 계속 사용하려는 기존 부트 환경이 있는 경우에는 ZFS 풀이나 파일 시스템의 이름을 변경하지 마십시오.	
<code>lucreate</code> 명령을 사용하기 전에 데이터 세트 등록 정보를 설정합니다.	Solaris Live Upgrade는 부트 환경에서 데이터 세트를 만들고 스왑 영역과 덤프 장치에서 ZFS 볼륨을 만들지만 기존 데이터 세트 등록 정보의 수정은 고려하지 않습니다. 즉 데이터 세트 등록 정보가 새 부트 환경에서 활성화되려면 <code>lucreate</code> 작업 이전에 이 등록 정보를 설정해야 합니다. 예를 들면 다음과 같습니다.	Solaris ZFS Administration Guide 의 “Introducing ZFS Properties”를 참조하십시오.
동일한 ZFS 루트 풀에서 ZFS 부트 환경을 만드는 경우 <code>lucreate</code> 명령 포함 또는 제외 옵션을 사용하여 내용을 사용자 정의할 수 없습니다.	<pre># zfs set compression=on rpool/ROOT</pre> <p>동일한 ZFS 루트 풀에서 부트 환경을 만드는 경우 <code>-f</code>, <code>-o</code>, <code>-y</code>, <code>-Y</code> 및 <code>-z</code> 옵션을 사용하여 기본 부트 환경에서 파일을 포함하거나 제외시킬 수 없습니다. 하지만 다음의 경우 이러한 옵션을 사용할 수 있습니다.</p> <ul style="list-style-type: none"> ■ UFS 파일 시스템 간 부트 환경 만들기 ■ UFS 파일 시스템에서 ZFS 루트 풀로 부트 환경 만들기 ■ ZFS 루트 풀에서 다른 ZFS 루트 풀로 부트 환경 만들기 	포함 또는 제외 옵션 사용에 대한 자세한 내용은 82 페이지 “부트 환경 만들기 및 내용 사용자 정의”를 참조하십시오.
Solaris Live Upgrade를 사용하여 루트가 아닌 ZFS 파일 시스템을 업그레이드할 수는 없습니다.		

추가 자원

이 장에 포함된 항목에 대한 추가 정보는 표 12-2에 나열된 자원을 참조하십시오.

표 12-2 추가 자원

자원	위치
ZFS 설치 계획에 대한 자세한 내용	Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획의 6 장, “ZFS 루트 파일 시스템 설치(계획)”
개요, 계획, 단계별 지침을 비롯한 ZFS 정보	Solaris ZFS Administration Guide
UFS 파일 시스템이 있는 시스템에서 Solaris Live Upgrade 사용	이 설명서의 제1부

ZFS 루트 풀의 부트 환경 만들기

이 장에서는 Solaris Live Upgrade를 사용할 때 ZFS 부트 환경을 만드는 방법에 대한 단계별 절차에 대해 설명합니다.

주 - UFS 파일 시스템에서 ZFS 루트 풀로의 마이그레이션이나 Solaris Live Upgrade를 사용한 ZFS 부트 환경 만들기는 **Solaris 10 10/08 릴리스**의 새로운 기능입니다. Solaris Live Upgrade를 UFS 파일 시스템이 있는 시스템에서 사용하려면 이 설명서의 제1부를 참조하십시오.

이 장에서는 다음 작업의 절차에 대해 설명합니다.

- 187 페이지 “UFS 파일 시스템을 ZFS 파일 시스템으로 마이그레이션”
- 194 페이지 “동일한 ZFS 루트 풀 내에 부트 환경 만들기”
- 198 페이지 “새 루트 풀에 부트 환경 만들기”
- 203 페이지 “현재 실행 중인 시스템 이외의 소스에서 부트 환경 만들기”
- 204 페이지 “ZFS 부트 환경으로 풀백”

비전역 영역이 설치되어 있는 경우 ZFS를 사용하는 절차는 14 장, “비전역 영역이 설치된 ZFS용 Solaris Live Upgrade”를 참조하십시오.

UFS 파일 시스템을 ZFS 파일 시스템으로 마이그레이션

이 절차에서는 UFS 파일 시스템을 ZFS 파일 시스템으로 마이그레이션하는 방법에 대해 설명합니다. 부트 환경을 만들면 중요 파일을 활성화 UFS 부트 환경에서 ZFS 루트 풀로 복사하는 방법이 제공됩니다. `lucreate` 명령은 중요 파일을 기존 ZFS 루트 풀에 있는 새 부트 환경으로 복사합니다. 사용자 정의(공유 가능) 파일 시스템은 복사되지 않으며 소스 UFS 부트 환경과 공유되지도 않습니다. 또한 `/swap`은 UFS 파일 시스템과 ZFS 루트 풀 사이에 공유되지 않습니다. 중요한 파일 시스템과 공유 가능한 파일 시스템에 대한 개요는 22 페이지 “파일 시스템 유형”을 참조하십시오.

▼ UFS 파일 시스템에서 ZFS 파일 시스템으로 마이그레이션하는 방법

주 - 활성 UFS 루트(/) 파일 시스템을 ZFS 루트 폴로 마이그레이션하려면 루트 폴 이름을 제공해야 합니다. 중요 파일 시스템은 루트 폴로 복사됩니다.

- 1 **Solaris Live Upgrade**를 처음으로 실행하기 전에 설치 매체에서 최신 **Solaris Live Upgrade** 패키지를 설치하고 **SunSolve** 정보 설명서 206844에 나열된 패치를 설치해야 합니다. **SunSolve** 웹 사이트에서 정보 설명서 206844(이전의 72099)를 검색합니다.

최신 패키지 및 패치를 설치하면 릴리스에 모든 최신 버그 수정과 새로운 기능이 포함됩니다. 새 부트 환경 만들기를 진행하기 전에 시스템과 관련된 모든 패치를 설치해야 합니다.

다음 하위 단계에서는 SunSolve 정보 설명서 206844에 나오는 단계에 대해 설명합니다.

주 - Solaris Live Upgrade를 사용하여 새 ZFS 부트 환경을 만들려면 최소 Solaris 10 10/08 릴리스가 설치되어 있어야 합니다. 이전 릴리스에는 작업을 수행할 수 있는 ZFS 및 Solaris Live Upgrade 소프트웨어가 없습니다.

- a. 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

- b. **Solaris Live Upgrade** 패키지를 제거하거나 추가하려면 SunSolve 웹 사이트에서 제공되는 정보 설명서 206844의 지침을 따르십시오.

세 개의 Solaris Live Upgrade 패키지 SUNWluu, SUNWlur 및 SUNWlucfg는 Solaris Live Upgrade를 사용한 업그레이드에 필요한 소프트웨어를 구성합니다. 이들 패키지는 기존 소프트웨어, 새 기능, 버그 수정 등을 포함합니다. 기존 패키지를 제거하지 않고 Solaris Live Upgrade를 사용하기 전에 시스템에 새 패키지를 설치한 경우에는 대상 릴리스로의 업그레이드가 실패합니다. SUNWlucfg 패키지는 **Solaris 10 8/07 릴리스부터** 지원되는 새로운 기능입니다. Solaris 10 8/07 이전 릴리스의 Solaris Live Upgrade 패키지를 사용하는 경우에는 이 패키지를 제거하지 않아도 됩니다.

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- c. 업그레이드 중인 릴리스에서 새 **Solaris Live Upgrade** 패키지를 설치합니다. 자세한 내용은 **58 페이지 "Solaris Live Upgrade 설치"**를 참조하십시오.

- d. **Solaris Live Upgrade**를 실행하기 전에 다음 패치를 설치해야 합니다. 이러한 패치에서는 최신 버그 수정과 릴리스의 새로운 기능을 설치합니다.

SunSolve을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 206844(이전의 72099)를 검색합니다.

- 패치를 로컬 디스크에 저장하려는 경우에는 /var/tmp/lupatches와 같은 디렉토리를 만들고 해당 디렉토리로 패치를 다운로드합니다.

- SunSolve 웹 사이트에서 패치 목록을 확인합니다.
- 패치 디렉토리로 변경합니다.

```
# cd /var/tmp/lupatches
```

- patchadd 명령을 사용하여 패치를 설치합니다.

```
# patchadd patch_id
```

*patch_id*는 패치 번호입니다. 공백을 사용하여 여러 패치 이름을 구분합니다.

주 - 패치는 정보 문서 206844에서 지정한 순서대로 적용해야 합니다.

- 필요한 경우 시스템을 재부트합니다. 특정 패치는 재부트해야만 적용됩니다.
x86에만 해당: 시스템을 재부트해야 하며, 그렇지 않으면 Solaris Live Upgrade가 실패합니다.

```
# init 6
```

이제 성공적으로 마이그레이션하는 데 필요한 패키지 및 패치가 설치되었습니다.

2 ZFS 루트 풀을 만듭니다.

ZFS 루트 풀은 부트 가능하고 업그레이드 가능하도록 단일 슬라이스에 있어야 합니다.

```
# zpool create rpool c0t1d0s5
```

rpool 만들려는 새 ZFS 루트 풀 이름을 지정합니다.

c0t1d0s5 디스크 슬라이스 *c0t1d0s5*에 새 루트 풀을 만듭니다.

새 루트 풀을 만드는 방법에 대한 자세한 내용은 [Solaris ZFS Administration Guide](#)를 참조하십시오.

3 UFS 루트(/) 파일 시스템을 새 ZFS 루트 풀로 마이그레이션합니다.

```
# lucreate [-c ufsBE] -n new-zfsBE -p rpool
```

-c ufsBE 이름 *ufsBE*를 현재 UFS 부트 환경에 할당합니다. 이 옵션은 필수 사항이 아니며 부트 환경을 처음 만들 때만 사용됩니다. 처음 **lucreate**를 실행할 때 **-c** 옵션을 생략하면 소프트웨어에서는 사용자의 기본 파일 이름을 사용합니다.

-n new-zfsBE 이름 *new-zfsBE*를 만들려는 부트 환경에 할당합니다. 이름은 시스템에서 고유해야 합니다.

-p rpool 새로 만든 ZFS 루트(/) 파일 시스템을 *rpool*에 정의된 ZFS 루트 풀에 배치합니다.

새 ZFS 부트 환경을 만들 때 약간 시간이 걸릴 수도 있습니다. UFS 파일 시스템 데이터는 ZFS 루트 풀로 복사됩니다. 비활성 부트 환경이 만들어지면 `luupgrade` 또는 `luactivate` 명령을 사용하여 새 ZFS 부트 환경을 업그레이드하거나 활성화할 수 있습니다.

4 (선택 사항) 부트 환경이 완전한지 확인합니다.

이 예에서 `lustatus` 명령은 부트 환경 만들기가 완료되어 부트 가능한지 보고합니다.

```
# lustatus
boot environment  Is      Active  Active  Can      Copy
Name             Complete Now      OnReboot Delete  Status
-----
ufsBE             yes     yes     yes     no      -
new-zfsBE        yes     no      no      yes     -
```

5 (선택 사항) 시스템에서 기본 데이터 세트 정보를 확인합니다.

`list` 명령은 시스템에 모든 데이터 집합 이름을 표시합니다. 이 예에서 `rpool`은 ZFS 풀 이름이고, `new-zfsBE`는 새로 만든 ZFS 부트 환경 이름입니다.

```
# zfs list
NAME                                USED AVAIL REFER MOUNTPOINT
rpool                               9.29G 57.6G 20K /rpool
rpool/ROOT                          5.38G 57.6G 18K /rpool/ROOT
rpool/ROOT/new-zfsBE                5.38G 57.6G 551M /tmp/.alt.luupdall.110034
rpool/dump                          1.95G - 1.95G -
rpool/swap                          1.95G - 1.95G -
```

새 부트 환경에 대해 나열된 마운트 지점은 `luactivate` 명령이 실행될 때까지 임시로 사용됩니다. `/dump` 및 `/swap` 볼륨은 원래의 UFS 부트 환경과 공유되지 않지만 ZFS 루트 풀과 루트 풀에 있는 부트 환경 내에서는 공유됩니다.

이제 새 부트 환경을 업그레이드하고 활성화할 수 있습니다. 예 13-1을 참조하십시오.

예 13-1 UFS 루트(/) 파일 시스템을 ZFS 루트 풀로 마이그레이션

이 예에서 새 ZFS 루트 풀인 `rpool`은 별도의 슬라이스 `C0t0d0s4`에서 만들어집니다. `lucreate` 명령은 현재 실행 중인 UFS 부트 환경 `c0t0d0`을 새 ZFS 부트 환경 `new-zfsBE`로 마이그레이션하고 새 부트 환경을 `rpool`에 배치합니다.

```
# zpool create rpool C0t0d0s4

# zfs list
NAME                                USED AVAIL REFER MOUNTPOINT
rpool                               9.29G 57.6G 20K /rpool
# lucreate -c c0t0d0 -n new-zfsBE -p rpool
Analyzing system configuration.
Current boot environment is named <c0t0d0>.
Creating initial configuration for primary boot environment <c0t0d0>.
```

```

The device </dev/dsk/c0t0d0> is not a root device for any boot
environment; cannot get BE ID.
PBE configuration successful: PBE name <c0t0d0> PBE Boot Device
</dev/dsk/c0t0d0>.
Comparing source boot environment <c0t0d0> file systems with the file
system(s) you specified for the new boot environment. Determining which
file systems should be in the new boot environment.
Updating boot environment description database on all BEs.
Updating system configuration files.
The device </dev/dsk/clt0d0s0> is not a root device for any boot
environment; cannot get BE ID.
Creating configuration for boot environment <new-zfsBE>.
Source boot environment is <c0t0d0>.
Creating boot environment <new-zfsBE>.
Creating file systems on boot environment <new-zfsBE>.
Creating <zfs> file system for </> in zone <global> on
<rpool/ROOT/new-zfsBE>.
Populating file systems on boot environment <new-zfsBE>.
Checking selection integrity.
Integrity check OK.
Populating contents of mount point </>.
Copying.
Creating shared file system mount points.
Creating compare databases for boot environment <zfsBE>.
Creating compare database for file system </>.
Making boot environment <zfsBE> bootable.
Creating boot_archive for /.alt.tmp.b-cBc.mnt
updating /.alt.tmp.b-cBc.mnt/platform/sun4u/boot_archive
Population of boot environment <new-zfsBE> successful.
Creation of boot environment <new-zfsBE> successful.

```

lustatus

boot environment Name	Is Complete	Active Now	Active OnReboot	Can Delete	Copy Status
c0t0d0	yes	yes	yes	no	-
new-zfsBE	yes	no	no	yes	-

zfs list

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool	9.29G	57.6G	20K	/rpool
rpool/ROOT	5.38G	57.6G	18K	/rpool/ROOT
rpool/ROOT/zfsBE	5.38G	57.6G	551M	
rpool/ROOT/new-zfsBE	5.38G	57.6G	551M	/tmp/.alt.luupdall.110034
rpool/dump	1.95G	-	1.95G	-
rpool/swap	1.95G	-	1.95G	-

이제 새 부트 환경을 업그레이드하거나 활성화할 수 있습니다.

이 예에서 새 부트 환경은 `-s` 옵션으로 지정된 위치에 저장된 이미지에서 `luupgrade` 명령을 사용하여 업그레이드됩니다.

```
# luupgrade -n zfsBE -u -s /net/install/export/s10/combined.s10
 51135 blocks
miniroot filesystem is <lofs>
Mounting miniroot at
</net/install/export/solaris_10/combined.solaris_10_wos
/Solaris_10/Tools/Boot>
Validating the contents of the media
</net/install/export/s10/combined.s10>.
The media is a standard Solaris media.
The media contains an operating system upgrade image.
The media contains Solaris version <10_1008>.
Constructing upgrade profile to use.
Locating the operating system upgrade program.
Checking for existence of previously scheduled Live
Upgrade requests.
Creating upgrade profile for BE <zfsBE>.
Determining packages to install or upgrade for BE <zfsBE>.
Performing the operating system upgrade of the BE <zfsBE>.
CAUTION: Interrupting this process may leave the boot environment
unstable or unbootable.
Upgrading Solaris: 100% completed
Installation of the packages from this media is complete.
Adding operating system patches to the BE <zfsBE>.
The operating system patch installation is complete.
INFORMATION: The file /var/sadm/system/logs/upgrade_log on boot
environment <zfsBE> contains a log of the upgrade operation.
INFORMATION: The file var/sadm/system/data/upgrade_cleanup on boot
environment <zfsBE> contains a log of cleanup operations required.
INFORMATION: Review the files listed above. Remember that all
of the files are located on boot environment <zfsBE>.
Before you activate boot environment <zfsBE>, determine if any
additional system maintenance is required or if additional media
of the software distribution must be installed.
The Solaris upgrade of the boot environment <zfsBE> is complete.
```

새 부트 환경을 만든 후에는 언제든지 활성화할 수 있습니다.

```
# luactivate new-zfsBE
*****
```

```
The target boot environment has been activated. It will be used when you
reboot. NOTE: You MUST NOT USE the reboot, halt, or uadmin commands. You
MUST USE either the init or the shutdown command when you reboot. If you
do not use either init or shutdown, the system will not boot using the
```

target BE.

In case of a failure while booting to the target BE, the following process needs to be followed to fallback to the currently working boot environment:

1. Enter the PROM monitor (ok prompt).
2. Change the boot device back to the original boot environment by typing:

```
setenv boot-device /pci@1f,0/pci@1/scsi@4,1/disk@2,0:a
```

3. Boot to the original boot environment by typing:

```
boot
```

Modifying boot archive service

Activation of boot environment <new-zfsBE> successful.

ZFS 부트 환경으로 시스템을 재부트합니다.

```
# init 6
```

```
# svc.startd: The system is coming down. Please wait.
```

```
svc.startd: 79 system services are now being stopped.
```

```
.
```

```
.
```

```
.
```

ZFS 저장소 풀은 UFS 부트 환경에서 자동으로 사용할 수 없으므로 UFS 부트 환경으로 풀백하면 ZFS 부트 환경에서 만들었던 ZFS 저장소 풀을 다시 가져와야 합니다. UFS 부트 환경으로 다시 전환하는 경우 다음 예와 유사한 메시지가 표시될 수 있습니다.

```
# luactivate c0t0d0
```

```
WARNING: The following files have changed on both the current boot environment <new-zfsBE> zone <global> and the boot environment to be activated <c0t0d0>:
```

```
  /etc/zfs/zpool.cache
```

```
INFORMATION: The files listed above are in conflict between the current boot environment <zfsBE> zone <global> and the boot environment to be activated <c0t0d0>. These files will not be automatically synchronized from the current boot environment <new-zfsBE> when boot environment <c0t0d0>
```

동일한 ZFS 루트 풀 내에 부트 환경 만들기

기존 ZFS 루트 풀이 있으며 해당 풀 내에서 새 ZFS 부트 환경을 만들려는 경우 다음 절차에서 제공하는 해당 단계를 참조하십시오. 비활성 부트 환경을 만든 후에는 편리한 시간에 새 부트 환경을 업그레이드하고 활성화할 수 있습니다. 동일한 풀 내에 부트 환경을 만드는 경우에는 -p 옵션이 필요 없습니다.

▼ 동일한 ZFS 루트 풀 내에서 ZFS 부트 환경을 만드는 방법

- 1 **Solaris Live Upgrade**를 처음으로 실행하기 전에 설치 매체에서 최신 **Solaris Live Upgrade** 패키지를 설치하고 **SunSolve** 정보 설명서 206844에 나열된 패치를 설치해야 합니다. **SunSolve** 웹 사이트에서 정보 설명서 206844(이전의 72099)를 검색합니다.

최신 패키지 및 패치를 설치하면 릴리스에 모든 최신 버그 수정과 새로운 기능이 포함됩니다. 새 부트 환경 만들기를 진행하기 전에 시스템과 관련된 모든 패치를 설치해야 합니다.

다음 하위 단계에서는 SunSolve 정보 설명서 206844에 나오는 단계에 대해 설명합니다.

주 - Solaris Live Upgrade를 사용하여 새 ZFS 부트 환경을 만들려면 최소 Solaris 10 10/08 릴리스가 설치되어 있어야 합니다. 이전 릴리스에는 작업을 수행할 수 있는 ZFS 및 Solaris Live Upgrade 소프트웨어가 없습니다.

a. 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

- b. **Solaris Live Upgrade** 패키지를 제거하거나 추가하려면 SunSolve 웹 사이트에서 제공되는 정보 설명서 206844의 지침을 따르십시오.

세 개의 Solaris Live Upgrade 패키지 SUNWluu, SUNWlur 및 SUNWlucfg는 Solaris Live Upgrade를 사용한 업그레이드에 필요한 소프트웨어를 구성합니다. 이들 패키지는 기존 소프트웨어, 새 기능, 버그 수정 등을 포함합니다. 기존 패키지를 제거하지 않고 Solaris Live Upgrade를 사용하기 전에 시스템에 새 패키지를 설치한 경우에는 대상 릴리스로의 업그레이드가 실패합니다. SUMWlucfg 패키지는 **Solaris 10 8/07 릴리스부터** 지원되는 새로운 기능입니다. Solaris 10 8/07 이전 릴리스의 Solaris Live Upgrade 패키지를 사용하는 경우에는 이 패키지를 제거하지 않아도 됩니다.

주 - SUMWlucfg 패키지는 **Solaris 10 8/07 릴리스부터** 지원되는 새로운 기능입니다. 이전 릴리스의 Solaris Live Upgrade 패키지를 사용하는 경우에는 이 패키지를 제거하지 않아도 됩니다.

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

c. 새 Solaris Live Upgrade 패키지를 설치합니다. 자세한 내용은 58 페이지 “Solaris Live Upgrade 설치”를 참조하십시오.

d. Solaris Live Upgrade를 실행하기 전에 다음 패치를 설치해야 합니다. 이러한 패치에서는 최신 버그 수정과 릴리스의 새로운 기능을 설치합니다.

SunSolve을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 206844(이전의 72099)를 검색합니다.

- 패치를 로컬 디스크에 저장하려는 경우에는 /var/tmp/lupatches와 같은 디렉토리를 만들고 해당 디렉토리로 패치를 다운로드합니다.
- SunSolve 웹 사이트에서 패치 목록을 확인합니다.
- 이 예에서처럼 패치 디렉토리를 변경합니다.

```
# cd /var/tmp/lupatches
```

- patchadd 명령을 사용하여 패치를 설치합니다.

```
# patchadd path-to-patches patch_id patch_id
```

*path-to-patches*는 패치 디렉토리의 경로(예: /var/tmp/lupatches)입니다. *patch_id*는 패치 번호입니다. 공백을 사용하여 여러 패치 이름을 구분합니다.

주 - 패치는 정보 문서 206844에서 지정한 순서대로 적용해야 합니다.

- 필요한 경우 시스템을 재부트합니다. 특정 패치는 재부트해야만 적용됩니다. **x86에만 해당:** 시스템을 재부트해야 하며, 그렇지 않으면 Solaris Live Upgrade가 실패합니다.

```
# init 6
```

이제 새 부트 환경을 성공적으로 만드는 데 필요한 패키지 및 패치가 설치되었습니다.

2 새 부트 환경을 만듭니다.

```
# lucreate [-c zfsBE] -n new-zfsBE
```

- c *zfsBE* 이름 *zfsBE*를 현재 부트 환경에 할당합니다. 이 옵션은 필수 사항이 아니며 부트 환경을 처음 만들 때만 사용됩니다. 처음 lucreate를 실행할 때 -c 옵션을 생략하면 소프트웨어에서는 기본 파일 이름을 사용합니다.
- n *new-zfsBE* 만들려는 부트 환경에 이름을 할당합니다. 이름은 시스템에서 고유해야 합니다.

새 부트 환경은 거의 즉시 만들어집니다. 현재 ZFS 루트 풀에 있는 각 데이터 세트의 스냅샷이 만들어지며, 클론이 각 스냅샷에서 만들어집니다. 스냅샷은 디스크 공간 측면에서 매우 효율적이며, 이 프로세스에서는 최소의 디스크 공간이 사용됩니다. 비활성 부트 환경이 만들어지면 `luupgrade` 또는 `luactivate` 명령을 사용하여 새 ZFS 부트 환경을 업그레이드하거나 활성화할 수 있습니다.

3 (선택 사항) 부트 환경이 완전한지 확인합니다.

`lustatus` 명령은 부트 환경 만들기가 완료되었고 부트 가능한지 보고합니다.

```
# lustatus
boot environment  Is      Active  Active  Can      Copy
Name             Complete Now     OnReboot Delete   Status
-----
zfsBE             yes     yes     yes     no       -
new-zfsBE         yes     no      no      yes      -
```

4 (선택 사항) 시스템에서 기본 데이터 세트 정보를 확인합니다.

이 예에서 ZFS 루트 풀은 `rpool`로 이름이 지정되며 `@` 기호는 스냅샷을 나타냅니다. 새 부트 환경의 마운트 지점은 `luactivate` 명령이 실행될 때까지 임시로 사용됩니다. `/dump` 및 `/swap` 볼륨은 ZFS 루트 풀과 루트 풀에 있는 부트 환경과 공유됩니다.

```
# zfs list
NAME                                USED AVAIL REFER MOUNTPOINT
rpool                                9.29G 57.6G 20K /rpool
rpool/ROOT                           5.38G 57.6G 18K /rpool/ROOT
rpool/ROOT/zfsBE                      5.38G 57.6G 551M
rpool/ROOT/zfsBE@new-zfsBE            66.5K - 551M -
rpool/ROOT/new-zfsBE                  85.5K 57.6G 551M /tmp/.alt.103197
rpool/dump                            1.95G - 1.95G -
rpool/swap                            1.95G - 1.95G -
```

이제 새 부트 환경을 업그레이드하고 활성화할 수 있습니다. 예 13-2를 참조하십시오.

예 13-2 동일한 ZFS 루트 풀 내에 부트 환경 만들기

다음 명령은 새 ZFS 부트 환경 `new-zfsBE`를 만듭니다. 부트 환경이 동일한 루트 풀 내에 만들어지므로 `-p` 옵션이 필요 없습니다.

```
# lucreate [-c zfsBE] -n new-zfsBE
Analyzing system configuration.
Comparing source boot environment <zfsBE> file systems with the file
system(s) you specified for the new boot environment. Determining which
file systems should be in the new boot environment.
Updating boot environment description database on all BEs.
Creating configuration for boot environment new-zfsBE.
Source boot environment is zfsBE.
Creating boot environment new-zfsBE.
```

```
Cloning file systems from boot environment zfsBE to create
boot environment new-zfsBE.
Creating snapshot for <rpool> on <rpool> Creating clone for <rpool>.
Setting canmount=noauto for <rpool> in zone <global> on <rpool>.
Population of boot environment zfsBE successful on <rpool>.
```

```
# lustatus
```

boot environment Name	Is Complete	Active Now	Active OnReboot	Can Delete	Copy Status
zfsBE	yes	yes	yes	no	-
new-zfsBE	yes	no	no	yes	-

```
# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool	9.29G	57.6G	20K	/rpool
rpool/ROOT	5.38G	57.6G	18K	/rpool/ROOT
rpool/ROOT/zfsBE	5.38G	57.6G	551M	
rpool/ROOT/zfsBE@new-zfsBE	66.5K	-	551M	-
rpool/ROOT/new-zfsBE	85.5K	57.6G	551M	/tmp/.alt.103197
rpool/dump	1.95G	-	1.95G	-
rpool/swap	1.95G	-	1.95G	-

이제 새 부트 환경을 업그레이드하고 활성화할 수 있습니다. ZFS 부트 환경의 업그레이드에 대한 예는 예 13-1을 참조하십시오. luupgrade 명령 사용에 대한 예는 5 장, “Solaris Live Upgrade를 통해 업그레이드(작업)”를 참조하십시오.

```
# luactivate new-zfsBE
```

```
*****
```

The target boot environment has been activated. It will be used when you reboot. NOTE: You MUST NOT USE the reboot, halt, or uadmin commands. You MUST USE either the init or the shutdown command when you reboot. If you do not use either init or shutdown, the system will not boot using the target BE.

```
*****
```

In case of a failure while booting to the target BE, the following process needs to be followed to fallback to the currently working boot environment:

1. Enter the PROM monitor (ok prompt).
2. Change the boot device back to the original boot environment by typing:

```
setenv boot-device /pci@1f,0/pci@1/scsi@4,1/disk@2,0:a
```

3. Boot to the original boot environment by typing:

```
boot
```

```
*****
```

```
Modifying boot archive service
Activation of boot environment <new-zfsBE> successful.
```

ZFS 부트 환경으로 시스템을 재부트합니다.

```
# init 6
# svc.startd: The system is coming down. Please wait.
svc.startd: 79 system services are now being stopped.
.
.
.
```

새 루트 풀에 부트 환경 만들기

기존 ZFS 루트 풀이 있지만 새 루트 풀에서 새 ZFS 부트 환경을 만들려는 경우 다음 절차에서 제공하는 해당 단계를 참조하십시오. 비활성 부트 환경을 만든 후에는 편리한 시간에 새 부트 환경을 업그레이드하고 활성화할 수 있습니다. 새 부트 환경을 배치하는 위치를 참조하려면 `-p` 옵션이 필요합니다. 기존 ZFS 루트 풀이 있어야 하며, 부트 가능하고 업그레이드 가능하도록 개별 슬라이스에 있어야 합니다.

▼ 새 ZFS 루트 풀에 부트 환경을 만드는 방법

- 1 **Solaris Live Upgrade**를 처음으로 실행하기 전에 설치 매체에서 최신 **Solaris Live Upgrade** 패키지를 설치하고 **SunSolve** 정보 설명서 206844에 나열된 패치를 설치해야 합니다. **SunSolve** 웹 사이트에서 정보 설명서 206844(이전의 72099)를 검색합니다.

최신 패키지 및 패치를 설치하면 릴리스에 모든 최신 버그 수정과 새로운 기능이 포함됩니다. 새 부트 환경 만들기를 진행하기 전에 시스템과 관련된 모든 패치를 설치해야 합니다.

다음 하위 단계에서는 SunSolve 정보 설명서 206844에 나오는 단계에 대해 설명합니다.

주 - Solaris Live Upgrade를 사용하여 새 ZFS 부트 환경을 만들려면 최소 Solaris 10 10/08 릴리스가 설치되어 있어야 합니다. 이전 릴리스에는 작업을 수행할 수 있는 ZFS 및 Solaris Live Upgrade 소프트웨어가 없습니다.

- a. 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

- b. Solaris Live Upgrade 패키지를 제거하거나 추가하려면 SunSolve 웹 사이트에서 제공되는 정보 설명서 206844의 지침을 따르십시오.

세 개의 Solaris Live Upgrade 패키지 SUNWluu, SUNWlur 및 SUNWlucfg는 Solaris Live Upgrade를 사용한 업그레이드에 필요한 소프트웨어를 구성합니다. 이들 패키지는 기존 소프트웨어, 새 기능, 버그 수정 등을 포함합니다. 기존 패키지를 제거하지 않고 Solaris Live Upgrade를 사용하기 전에 시스템에 새 패키지를 설치한 경우에는 대상 릴리스로의 업그레이드가 실패합니다. SUNWlucfg 패키지는 **Solaris 10 8/07 릴리스부터** 지원되는 새로운 기능입니다. Solaris 10 8/07 이전 릴리스의 Solaris Live Upgrade 패키지를 사용하는 경우에는 이 패키지를 제거하지 않아도 됩니다.

주 - SUNWlucfg 패키지는 **Solaris 10 8/07 릴리스부터** 지원되는 새로운 기능입니다. 이전 릴리스의 Solaris Live Upgrade 패키지를 사용하는 경우에는 이 패키지를 제거하지 않아도 됩니다.

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- c. 새 Solaris Live Upgrade 패키지를 설치합니다. 자세한 내용은 58 페이지 "[Solaris Live Upgrade 설치](#)"를 참조하십시오.

- d. Solaris Live Upgrade를 실행하기 전에 다음 패치를 설치해야 합니다. 이러한 패치에서는 최신 버그 수정과 릴리스의 새로운 기능을 설치합니다.

SunSolve을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 206844(이전의 72099)를 검색합니다.

- 패치를 로컬 디스크에 저장하려는 경우에는 /var/tmp/lupatches와 같은 디렉토리를 만들고 해당 디렉토리로 패치를 다운로드합니다.
- SunSolve 웹 사이트에서 패치 목록을 확인합니다.
- 이 예에서처럼 패치 디렉토리를 변경합니다.

```
# cd /var/tmp/lupatches
```

- patchadd 명령을 사용하여 패치를 설치합니다.

```
# patchadd path-to-patches patch_id patch_id
```

*path-to-patches*는 패치 디렉토리의 경로(예: /var/tmp/lupatches)입니다. *patch_id*는 패치 번호입니다. 공백을 사용하여 여러 패치 이름을 구분합니다.

주 - 패치는 정보 문서 206844에서 지정한 순서대로 적용해야 합니다.

- 필요한 경우 시스템을 재부트합니다. 특정 패치는 재부트해야만 적용됩니다.

x86에만 해당: 시스템을 재부트해야 하며, 그렇지 않으면 Solaris Live Upgrade가 실패합니다.

```
# init 6
```

이제 성공적으로 마이그레이션하는 데 필요한 패키지 및 패치가 설치되었습니다.

2 ZFS 루트 풀을 만듭니다.

ZFS 루트 풀은 부트 가능하고 업그레이드 가능하도록 단일 슬라이스에 있어야 합니다.

```
# zpool create rpool2 c0t1d0s5
```

rpool2 새 ZFS 루트 풀의 이름입니다.

c0t1d0s5 부트 가능 슬라이스 *c0t1d0s5*에 *rpool2*를 배치하도록 지정합니다.

새 루트 풀을 만드는 방법에 대한 자세한 내용은 [Solaris ZFS Administration Guide](#)를 참조하십시오.

3 새 부트 환경을 만듭니다.

```
# lucreate [-c zfsBE] -n new-zfsBE -p rpool2
```

-c zfsBE 이름 *zfsBE*를 현재 ZFS 부트 환경에 할당합니다.

-n new-zfsBE 만들려는 부트 환경에 이름을 할당합니다. 이름은 시스템에서 고유해야 합니다.

-p rpool2 새로 만든 ZFS 루트 부트 환경을 *rpool2*에 정의된 ZFS 루트 풀에 배치합니다.

새 ZFS 부트 환경을 만들 때 약간 시간이 걸릴 수도 있습니다. 파일 시스템 데이터는 새 ZFS 루트 풀로 복사됩니다. 비활성 부트 환경이 만들어지면 `luupgrade` 또는 `luactivate` 명령을 사용하여 새 ZFS 부트 환경을 업그레이드하거나 활성화할 수 있습니다.

4 (선택 사항) 부트 환경이 완전한지 확인합니다.

`lustatus` 명령은 부트 환경 만들기가 완료되었고 부트 가능한지 보고합니다.

```
# lustatus
```

boot environment Name	Is Complete	Active Now	Active OnReboot	Can Delete	Copy Status
zfsBE		yes	yes	yes	no -
new-zfsBE		yes	no	no	yes -

5 (선택 사항) 시스템에서 기본 데이터 세트 정보를 확인합니다.

다음 예에서는 시스템에 모든 데이터 세트 이름을 표시합니다. 새 부트 환경에 대해 나열된 마운트 지점은 `luactivate` 명령이 실행될 때까지 임시로 사용됩니다. 새 부트 환경은 볼륨 `rpool2/dump` 및 `rpool2/swap`을 `rpool2` ZFS 부트 환경과 공유합니다.

```
# zfs list
NAME                                USED    AVAIL    REFER    MOUNTPOINT
rpool2                              9.29G   57.6G    20K      /rpool2
rpool2/ROOT/                        5.38G   57.6G    18K      /rpool2/ROOT
rpool2/ROOT/new-zfsBE               5.38G   57.6G    551M     /tmp/.new.luupdall.109859
rpool2/dump                         3.99G   -        3.99G    -
rpool2/swap                         3.99G   -        3.99G    -
rpool                               9.29G   57.6G    20K      /.new.lulib.rs.109262
rpool/ROOT                          5.46G   57.6G    18K      legacy
rpool/ROOT/zfsBE                   5.46G   57.6G    551M
rpool/dump                          3.99G   -        3.99G    -
rpool/swap                          3.99G   -        3.99G    -
```

이제 새 부트 환경을 업그레이드하고 활성화할 수 있습니다. 예 13-3을 참조하십시오.

예 13-3 새 루트 풀에 부트 환경 만들기

이 예에서 새 ZFS 루트 풀인 `rpool`은 별도의 슬라이스 `C0t2d0s5`에서 만들어집니다. `lucreate` 명령은 새 ZFS 부트 환경 `new-zfsBE`를 만듭니다. 부트 환경이 서로 다른 루트 풀 내에 만들어지므로 `-p` 옵션이 필요합니다.

```
# zpool create rpool C0t1d0s5
# zfs list
NAME                                USED    AVAIL    REFER    MOUNTPOINT
rpool2                              9.29G   57.6G    20K      /rpool2
rpool                               9.29G   57.6G    20K      /.new.lulib.rs.109262
rpool/ROOT                          5.46G   57.6G    57.6G    18K      legacy
rpool/ROOT/zfsBE                   5.46G   57.6G    551M
rpool/dump                          3.99G   -        3.99G    -
rpool/swap                          3.99G   -        3.99G    -
```

```
# lucreate -c rpool -n new-zfsBE -p rpool2
Analyzing system configuration.
Current boot environment is named <rpool>.
Creating initial configuration for primary boot environment <rpool>.
The device </dev/dsk/c0t0d0> is not a root device for any
boot environment; cannot get BE ID.
PBE configuration successful: PBE name <rpool> PBE Boot
Device </dev/dsk/rpool>.
Comparing source boot environment <rpool> file systems with
the file system(s) you specified for the new boot environment.
Determining which file systems should be in the new boot environment.
```

```

Updating boot environment description database on all BEs.
Updating system configuration files.
The device </dev/dsk/clt0d0s0> is not a root device for any
boot environment; cannot get BE ID.
Creating configuration for boot environment <new-zfsBE>.
Source boot environment is <rpool>.
Creating boot environment <new-zfsBE>.
Creating file systems on boot environment <new-zfsBE>.
Creating <zfs> file system for </> in zone <global> on
<rpool2/ROOT/new-zfsBE>.
Populating file systems on boot environment <new-zfsBE>.
Checking selection integrity.
Integrity check OK.
Populating contents of mount point </>.
Copying.
Creating shared file system mount points.
Creating compare databases for boot environment <zfsBE>.
Creating compare database for file system </>.
Making boot environment <new-zfsBE> bootable.
Creating boot_archive for /.alt.tmp.b-cBc.mnt
updating /.alt.tmp.b-cBc.mnt/platform/sun4u/boot_archive
Population of boot environment <new-zfsBE> successful.
Creation of boot environment <new-zfsBE> successful.

```

lustatus

boot environment Name	Is Complete	Active Now	Active OnReboot	Can Delete	Copy Status
zfsBE	yes	yes	yes	no	-
new-zfsBE	yes	no	no	yes	-

zfs list

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool2	9.29G	57.6G	20K	/rpool2
rpool2/ROOT/	5.38G	57.6G	18K	/rpool2/ROOT
rpool2/ROOT/new-zfsBE	5.38G	57.6G	551M	/tmp/.new.luupdall.109859
rpool2/dump	3.99G	-	3.99G	-
rpool2/swap	3.99G	-	3.99G	-
rpool	9.29G	57.6G	20K	/.new.lulib.rs.109262
rpool/ROOT	5.46G	57.6G	18K	legacy
rpool/ROOT/zfsBE	5.46G	57.6G	551M	
rpool/dump	3.99G	-	3.99G	-
rpool/swap	3.99G	-	3.99G	-

현재 실행 중인 시스템 이외의 소스에서 부트 환경 만들기

기존 ZFS 루트 풀이 있거나 현재 활성화 부트 환경으로 사용되지 않는 UFS 부트 환경이 있는 경우 다음 예를 사용하여 이 부트 환경에서 새 ZFS 부트 환경을 만들 수 있습니다. 새 ZFS 부트 환경을 만든 후에는 편리한 시간에 새 부트 환경을 업그레이드하고 활성화할 수 있습니다.

부트 환경을 현재 실행 중인 시스템 이외의 소스에서 만들려는 경우 `lucreate` 명령을 `-s` 옵션과 함께 사용해야 합니다. `-s` 옵션은 UFS 파일 시스템의 경우와 동일하게 작동합니다. `-s` 옵션은 대체 루트(/) 파일 시스템의 경로를 제공합니다. 이 대체 루트(/) 파일 시스템은 새 ZFS 루트 풀을 만들기 위한 소스입니다. 대체 루트는 UFS(/) 루트 파일 시스템이거나 ZFS 루트 풀일 수 있습니다. 복사 프로세스는 사용 중인 시스템에 따라 시간이 걸릴 수 있습니다.

다음 예에서는 다른 ZFS 루트 풀에서 부트 환경을 만들 때 `-s` 옵션이 사용되는 방법을 보여 줍니다.

예 13-4 현재 실행 중인 시스템 이외의 소스에서 부트 환경을 만드는 방법

다음 명령은 기존 ZFS 루트 풀에서 새 ZFS 루트 풀을 만듭니다. `-n` 옵션은 만들려는 부트 환경에 이름 `new-zfsBE`를 할당합니다. `-s` 옵션은 부트 환경 `rpool3`이 현재 실행 중인 부트 환경 대신 복사본의 소스로 사용되도록 지정합니다. `-p` 옵션은 `rpool2`에 새 부트 환경을 배치하도록 지정합니다.

```
# lucreate -n new-zfsBE -s rpool3 -p rpool2
# lustatus
```

boot environment Name	Is Complete	Active Now	Active OnReboot	Can Delete	Copy Status
zfsBE	yes	yes	yes	no	-
zfsBE2	yes	no	no	yes	-
zfsBE3	yes	no	no	yes	-
new-zfsBE	yes	no	no	yes	-

```
# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool2	9.29G	57.6G	20K	/rpool2
rpool2/ROOT/	5.38G	57.6G	18K	/rpool2/ROOT
rpool2/ROOT/new-zfsBE	5.38G	57.6G	551M	/tmp/.new.luupdall.109859
rpool2/dump	3.99G	-	3.99G	-
rpool2/swap	3.99G	-	3.99G	-
rpool3	9.29G	57.6G	20K	/rpool2
rpool3/ROOT/	5.38G	57.6G	18K	/rpool2/ROOT
rpool3/ROOT/zfsBE3	5.38G	57.6G	551M	/tmp/.new.luupdall.109859
rpool3/dump	3.99G	-	3.99G	-
rpool3/swap	3.99G	-	3.99G	-

예 13-4 현재 실행 중인 시스템 이외의 소스에서 부트 환경을 만드는 방법 (계속)

pool	9.29G	57.6G	20K	/.new.lulib.rs.109262
rpool/ROOT	5.46G	57.6G	18K	legacy
rpool/ROOT/zfsBE	5.46G	57.6G	551M	
rpool/dump	3.99G	-	3.99G	-
rpool/swap	3.99G	-	3.99G	-

이제 새 부트 환경을 업그레이드하고 활성화할 수 있습니다.

ZFS 부트 환경으로 폴백

업그레이드 후 오류가 감지되거나 응용 프로그램이 업그레이드된 구성 요소와 호환되지 않는 경우 `luactivate` 명령을 사용하여 원래의 부트 환경으로 폴백할 수 있습니다.

UFS 부트 환경에서 ZFS 루트 풀로 마이그레이션한 다음 UFS 부트 환경으로 폴백하도록 결정하면 ZFS 부트 환경에서 만들었던 ZFS 저장소 풀을 다시 가져와야 합니다. 이 ZFS 저장소 풀은 UFS 부트 환경에서 자동으로 사용할 수 없습니다. UFS 부트 환경으로 다시 전환하는 경우 다음 예와 유사한 메시지가 표시될 수 있습니다.

```
# luactivate c0t0d0
```

```
WARNING: The following files have changed on both the current boot
environment <new-ZFSbe> zone <global> and the boot environment
to be activated <c0t0d0>: /etc/zfs/zpool.cache
```

```
INFORMATION: The files listed above are in conflict between the current
boot environment <ZFSbe> zone <global> and the boot environment to be
activated <c0t0d0>. These files will not be automatically synchronized
from the current boot environment <new-ZFSbe> when boot
environment <c0t0d0>
```

원래 부트 환경으로 폴백에 대한 예는 6 장, “오류 복구: 원래 부트 환경으로 폴백(작업)”을 참조하십시오.

추가 자원

이 장에 포함된 항목에 대한 추가 정보는 표 13-1에 나열된 자원을 참조하십시오.

표 13-1 추가 자원

자원	위치
개요, 계획, 단계별 지침을 비롯한 ZFS 정보	Solaris ZFS Administration Guide

표 13-1 추가 자원 (계속)

자원	위치
UFS 파일 시스템이 있는 시스템에서 Solaris Live Upgrade 사용	이 설명서의 제1부

비전역 영역이 설치된 ZFS용 Solaris Live Upgrade

이 장에서는 UFS(/) 루트 파일 시스템을 ZFS 루트 풀로 마이그레이션하는 작업에 대한 개요와 단계별 절차를 제공합니다.

- 207 페이지 “비전역 영역이 설치된 시스템에 ZFS 부트 환경 만들기(개요 및 계획)”
- 208 페이지 “비전역 영역이 설치된 UFS 루트(/) 파일 시스템에서 ZFS 루트 풀로 마이그레이션(작업)”

주 - UFS 루트(/) 파일 시스템에서 ZFS 루트 풀로의 마이그레이션이나 Solaris Live Upgrade를 사용한 ZFS 부트 환경 만들기는 **Solaris 10 10/08 릴리스**의 새로운 기능입니다. UFS 파일 시스템에서 Solaris Live Upgrade를 수행하면 명령줄 매개 변수와 Solaris Live Upgrade의 작동이 둘 다 변경되지 않는 상태로 유지됩니다. Solaris Live Upgrade를 UFS 파일 시스템이 있는 시스템에서 수행하려면 이 설명서의 **제1부**를 참조하십시오.

비전역 영역이 설치된 시스템에 ZFS 부트 환경 만들기(개요 및 계획)

Solaris Live Upgrade를 사용하여 비전역 영역이 설치된 UFS 루트(/) 파일 시스템을 ZFS 루트 풀로 마이그레이션할 수 있습니다. 파일 시스템과 연관된 모든 비전역 영역은 새 부트 환경으로도 복사됩니다. 다음 비전역 영역 마이그레이션 시나리오가 지원됩니다.

사전 마이그레이션 루트 파일 시스템 및 영역 결합

UFS 파일 시스템에서 비전역 영역 루트 디렉토리가 있는 UFS 루트 파일 시스템

사후 마이그레이션 루트 파일 시스템 및 영역 결합

ZFS 루트 풀에서 비전역 영역 루트 디렉토리가 있는 UFS 루트 파일 시스템

ZFS 루트 풀에서 비전역 영역 루트 디렉토리가 있는 ZFS 루트 풀

사전 마이그레이션 루트 파일 시스템 및 영역 결합	사후 마이그레이션 루트 파일 시스템 및 영역 결합
	UFS 파일 시스템에서 비전역 영역 루트 디렉토리가 있는 ZFS 루트폴
ZFS 루트폴에서 비전역 영역 루트가 있는 UFS 루트 파일 시스템	ZFS 루트폴에서 비전역 영역 루트가 있는 ZFS 루트폴
	ZFS 루트폴에서 비전역 영역 루트가 있는 UFS 루트 파일 시스템
ZFS 루트폴에서 비전역 영역 루트 디렉토리가 있는 ZFS 루트폴	ZFS 루트폴에서 비전역 영역 루트 디렉토리가 있는 ZFS 루트폴

UFS 루트(/) 파일 시스템과 비전역 영역이 설치된 시스템에서 비전역 영역이 UFS - ZFS 마이그레이션의 일부로 공유되지 않는 파일 시스템에 있는 경우 해당 영역이 마이그레이션됩니다. 그렇지 않고 동일한 ZFS 풀에서 업그레이드하는 경우에는 영역이 복제됩니다. 비전역 영역이 공유 UFS 파일 시스템에 있는 경우 다른 ZFS 루트폴로 마이그레이션하려면 먼저 이전 Solaris 릴리스에서와 같이 비전역 영역을 업그레이드해야 합니다.

- ZFS 루트폴로 마이그레이션할 때 계획에 대한 자세한 내용은 [183 페이지 “Solaris Live Upgrade 사용 시 시스템 요구 사항 및 제한”](#)을 참조하십시오.
- ZFS 및 비전역 영역의 제한 사항에 대한 자세한 내용은 [Solaris ZFS Administration Guide](#)의 [“Using ZFS on a Solaris System With Zones Installed”](#)를 참조하십시오.

비전역 영역이 설치된 UFS 루트(/) 파일 시스템에서 ZFS 루트폴로 마이그레이션(작업)

이 장에서는 비전역 영역이 설치된 시스템에서 UFS 루트(/) 파일 시스템을 ZFS 루트폴로 마이그레이션하는 단계별 지침을 제공합니다. UFS 파일 시스템의 공유 파일 시스템에는 비전역 영역이 없습니다.

▼ 비전역 영역이 설치된 시스템에서 UFS 파일 시스템을 ZFS 파일 시스템으로 마이그레이션하는 방법

lucreate 명령은 UFS 루트(/) 파일 시스템에서 ZFS 루트폴의 부트 환경을 만듭니다. ZFS 루트폴은 lucreate 작업 전에 있어야 하며, 업그레이드 가능하고 부트 가능하도록 하려면 전체 디스크가 아닌 슬라이스로 만들어야 합니다. 이 절차에서는 UFS 루트(/) 파일 시스템과 연결된 기존의 비전역 영역을 ZFS 루트폴에 있는 새 부트 환경에 복사하는 방법에 대해 보여 줍니다.

다음 예에서는 기존 비전역 영역 myzone에 UFS 루트(/) 파일 시스템의 비전역 영역 루트가 있으며, 영역 zzone에는 기존 ZFS 저장소 풀 pool에 있는 ZFS 파일 시스템의 영역 루트가 있습니다. Solaris Live Upgrade는 UFS 부트 환경 c2t2d0s0을 ZFS 부트 환경 zfs2BE로 마이그레이션하는 데 사용됩니다. UFS 기반 myzone 영역은 Solaris Live Upgrade 작업 전에 만든 새 ZFS 저장소 풀 mpool로 마이그레이션합니다. ZFS 기반 비전역 영역 zzone은 복제되지만 ZFS 풀 pool에서 유지되며 새 zfs2BE 부트 환경으로 마이그레이션됩니다.

1 Solaris Live Upgrade를 맨 처음 수행할 때 다음 단계를 수행합니다.

주 - Solaris Live Upgrade를 사용하여 새 ZFS 부트 환경을 만들려면 최소 **Solaris 10 10/08 릴리스**가 설치되어 있어야 합니다. 이전 릴리스에는 작업을 수행할 수 있는 ZFS 및 Solaris Live Upgrade 소프트웨어가 없습니다.

- a. 필요한 경우 시스템에서 기존 Solaris Live Upgrade 패키지를 제거합니다. 새 릴리스로 업그레이드하려면 해당 릴리스의 패키지를 설치해야 합니다.

세 개의 Solaris Live Upgrade 패키지 SUNWluu, SUNWlur 및 SUNWlucfg는 Solaris Live Upgrade를 사용한 업그레이드에 필요한 소프트웨어를 구성합니다. 이들 패키지는 기존 소프트웨어, 새 기능, 버그 수정 등을 포함합니다. 기존 패키지를 제거하지 않고 Solaris Live Upgrade를 사용하기 전에 시스템에 새 패키지를 설치한 경우에는 대상 릴리스로의 업그레이드가 실패합니다.

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- b. 업그레이드 중인 릴리스에서 새 Solaris Live Upgrade 패키지를 설치합니다. 자세한 내용은 58 페이지 "**Solaris Live Upgrade 설치**"를 참조하십시오.

- c. Solaris Live Upgrade를 설치하거나 실행하기 전에 다음 패치를 설치해야 합니다. 이러한 패치에서는 최신 버그 수정과 릴리스의 새로운 기능을 설치합니다.

SunSolve을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 206844(이전의 72099)를 검색합니다.

- 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
- 패치를 로컬 디스크에 저장하려는 경우에는 /var/tmp/lupatches와 같은 디렉토리를 만들고 해당 디렉토리로 패치를 다운로드합니다.
- SunSolve 웹 사이트에서 패치 목록을 확인합니다.
- 패치 디렉토리로 변경합니다.

```
# cd /var/tmp/lupatches
```

- patchadd 명령을 사용하여 패치를 설치합니다.

```
# patchadd patch_id
```

patch_id는 패치 번호입니다. 공백을 사용하여 여러 패치 이름을 구분합니다.

주 - 패치는 정보 문서 206844에서 지정한 순서대로 적용해야 합니다.

- 필요한 경우 시스템을 재부트합니다. 특정 패치는 재부트해야만 적용됩니다.
x86에만 해당: 시스템을 재부트해야 하며, 그렇지 않으면 Solaris Live Upgrade가 실패합니다.

init 6

2 ZFS 루트 풀을 만듭니다.

ZFS 루트 풀은 부트 가능하고 업그레이드 가능하도록 단일 슬라이스에 있어야 합니다.

```
# zpool create rpool c3t0d0s0
```

이 예에서 만들려는 새 ZFS의 이름은 *rpool*입니다. 풀은 부트 가능한 슬라이스 *c3t0d0s0*에서 만들어집니다.

새 루트 풀을 만드는 방법에 대한 자세한 내용은 [Solaris ZFS Administration Guide](#)를 참조하십시오.

3 UFS 루트(/) 파일 시스템을 새 ZFS 루트 풀로 마이그레이션합니다.

```
# lucreate [-c ufsBE] -n new-zfsBE -p rpool
```

-c *ufsBE* 이름 *ufsBE*를 현재 UFS 부트 환경에 할당합니다. 이 옵션은 필수 사항이 아니며 부트 환경을 처음 만들 때만 사용됩니다. 처음 **lucreate**를 실행할 때 -c 옵션을 생략하면 소프트웨어에서는 사용자의 기본 파일 이름을 사용합니다.

-n *new-zfsBE* 이름 *new-zfsBE*를 만들려는 부트 환경에 할당합니다. 이름은 시스템에서 고유해야 합니다.

-p *rpool* 새로 만든 ZFS 루트(/) 파일 시스템을 *rpool*에 정의된 ZFS 루트 풀에 배치합니다.

공유되지 않는 모든 비전역 영역은 중요 파일 시스템과 함께 새 부트 환경으로 복사됩니다. 새 ZFS 부트 환경을 만들 때 약간 시간이 걸릴 수도 있습니다. UFS 파일 시스템 데이터는 ZFS 루트 풀로 복사됩니다. 비활성 부트 환경이 만들어지면 **luupgrade** 또는 **luactivate** 명령을 사용하여 새 ZFS 부트 환경을 업그레이드하거나 활성화할 수 있습니다.

4 (선택 사항) 부트 환경이 완전한지 확인합니다.

lustatus 명령은 부트 환경 만들기가 완료되었고 부트 가능한지 보고합니다.

```
# lustatus
```

boot environment	Is	Active	Active	Can	Copy
Name	Complete	Now	OnReboot	Delete	Status

ufsBE	yes	yes	yes	no	-
new-zfsBE	yes	no	no	yes	-

5 (선택 사항) 시스템에서 기본 데이터 세트 정보를 확인합니다.

`list` 명령은 시스템에 모든 데이터 집합 이름을 표시합니다. 이 예에서 `rpool`은 ZFS 풀 이름이고, `new-zfsBE`는 새로 만든 ZFS 부트 환경 이름입니다.

```
# zfs list
NAME                                USED  AVAIL  REFER  MOUNTPOINT
rpool                               9.29G  57.6G   20K    /rpool
rpool/ROOT                          5.38G  57.6G   18K    /rpool/ROOT
rpool/ROOT/new-zfsBE                5.38G  57.6G   55M    /tmp/.alt.luupdall.110034
rpool/dump                          1.95G  -      1.95G  -
rpool/swap                          1.95G  -      1.95G  -
```

새 부트 환경에 대해 나열된 마운트 지점은 `luactivate` 명령이 실행될 때까지 임시로 사용됩니다. `/dump` 및 `/swap` 볼륨은 원래의 UFS 부트 환경과 공유되지 않지만 ZFS 루트 풀과 루트 풀에 있는 부트 환경 내에서는 공유됩니다.

예 14-1 비전역 영역이 설치된 UFS 루트(/) 파일 시스템에서 ZFS 루트 풀로 마이그레이션

다음 예에서는 기존 비전역 영역 `myzone`에 UFS 루트(/) 파일 시스템의 비전역 영역 루트가 있으며, 영역 `zzone`에는 기존 ZFS 저장소 풀 `pool`에 있는 ZFS 파일 시스템의 영역 루트가 있습니다. Solaris Live Upgrade는 UFS 부트 환경 `c2t2d0s0`을 ZFS 부트 환경 `zfs2BE`로 마이그레이션하는 데 사용됩니다. UFS 기반 `myzone` 영역은 Solaris Live Upgrade 작업 전에 만든 새 ZFS 저장소 풀 `mpool`로 마이그레이션합니다. ZFS 기반 비전역 영역 `zzone`은 복제되지만 ZFS 풀 `pool`에서 유지되며 새 `zfs2BE` 부트 환경으로 마이그레이션됩니다.

```
# zoneadm list -iv
ID NAME          STATUS  PATH                BRAND  IP
0  global         running /                    native shared
-  myzone         installed /zones/myzone      native shared
-  zzone          installed /pool/zones         native shared

# zpool create mpool mirror c3t0d0s0 c4t0d0s0
# lucreate -c c1t2d0s0 -n zfs2BE -p mpool
Analyzing system configuration.
No name for current boot environment.
Current boot environment is named <c1t2d0s0>.
Creating initial configuration for primary boot environment <c1t2d0s0>.
The device </dev/dsk/c1t2d0s0> is not a root device for any
boot environment; cannot get BE ID.
PBE configuration successful: PBE name <c1t2d0s0> PBE Boot Device
</dev/dsk/c1t2d0s0>.
```

```

Comparing source boot environment <clt2d0s0> file systems with the file
system(s) you specified for the new boot environment. Determining which
file systems should be in the new boot environment.
Updating boot environment description database on all BEs.
Updating system configuration files.
The device </dev/dsk/clt0d0s0> is not a root device for any boot
environment; cannot get BE ID.
Creating configuration for boot environment <zfsBE>.
Source boot environment is <clt2d0s0>.
Creating boot environment <zfsBE>.
Creating file systems on boot environment <zfsBE>.
Creating <zfs> file system for </> in zone <global> on <rpool/ROOT/zfsBE>.
Populating file systems on boot environment <zfsBE>.
Checking selection integrity.
Integrity check OK.
Populating contents of mount point </>.
Copying.
Creating shared file system mount points.
Creating compare databases for boot environment <zfsBE>.
Creating compare database for file system </>.
Making boot environment <zfsBE> bootable.
Creating boot_archive for /.alt.tmp.b-cBc.mnt
updating /.alt.tmp.b-cBc.mnt/platform/sun4u/boot_archive
Population of boot environment <zfsBE> successful.
Creation of boot environment <zfsBE> successful.
    
```

lucreate 작업이 완료되면 lustatus 명령을 사용하여 이 예에서와 같이 부트 환경 상태를 살펴봅니다.

```
# lustatus
```

Boot Environment Name	Is Complete	Active Now	Active On Reboot	Can Delete	Copy Status
clt2d0s0	yes	yes	yes	no	-
zfsBE	yes	no	no	yes	-

```
# zoneadm list -iv
```

ID	NAME	STATUS	PATH	BRAND	IP
0	global	running	/	native	shared
-	myzone	installed	/zones/myzone	native	shared
-	zzone	installed	/pool/zones	native	shared

다음으로 luactivate 명령을 사용하여 새 ZFS 부트 환경을 활성화합니다. 예를 들면 다음과 같습니다.

```
# luactivate zfsBE
```

```
*****
```

The target boot environment has been activated. It will be used when you reboot. NOTE: You MUST NOT USE the reboot, halt, or uadmin commands. You MUST USE either the init or the shutdown command when you reboot. If you do not use either init or shutdown, the system will not boot using the target BE.

In case of a failure while booting to the target BE, the following process needs to be followed to fallback to the currently working boot environment:

1. Enter the PROM monitor (ok prompt).
2. Change the boot device back to the original boot environment by typing:

```
setenv boot-device /pci@1f,0/pci@1/scsi@4,1/disk@2,0:a
```

3. Boot to the original boot environment by typing:

```
boot
```

```
Modifying boot archive service
Activation of boot environment <ZFSbe> successful.
```

ZFS BE로 시스템을 재부트합니다.

```
# init 6
# svc.startd: The system is coming down. Please wait.
svc.startd: 79 system services are now being stopped.
.
.
.
```

이 예에서와 같이 새 부트 환경과 마이그레이션된 영역의 상태를 확인합니다.

```
# lustatus
```

Boot Environment Name	Is Complete	Active Now	Active On Reboot	Can Delete	Copy Status
clt2d0s0	yes	yes	yes	no	-
zfsBE	yes	no	no	yes	-

ZFS 저장소 풀은 UFS 부트 환경에서 자동으로 사용할 수 없으므로 UFS 부트 환경으로 폴백하면 ZFS 부트 환경에서 만들었던 ZFS 저장소 풀을 다시 가져와야 합니다. UFS 부트 환경으로 다시 전환하는 경우 다음과 유사한 메시지가 표시될 수 있습니다.

```
# luactivate c1t2d0s0
WARNING: The following files have changed on both the current boot
environment <ZFSbe> zone <global> and the boot environment to be activated <c1t2d0s0>:
/etc/zfs/zpool.cache
INFORMATION: The files listed above are in conflict between the current
boot environment <ZFSbe> zone <global> and the boot environment to be
activated <c1t2d0s0>. These files will not be automatically synchronized
from the current boot environment <ZFSbe> when boot environment <c1t2d0s0>
```

추가 자원

이 장에 포함된 항목에 대한 추가 정보는 표 14-1에 나열된 자원을 참조하십시오.

표 14-1 추가 자원

자원	위치
개요, 계획, 단계별 지침을 비롯한 비전역 영역 정보	System Administration Guide: Solaris Containers-Resource Management and Solaris Zones
개요, 계획, 단계별 지침을 비롯한 ZFS 정보	Solaris ZFS Administration Guide
UFS 파일 시스템이 있는 시스템에서 Solaris Live Upgrade 사용 정보	이 설명서의 제1부, 8 장, “비전역 영역이 설치된 시스템에서 Solaris OS 업그레이드”

3 부록

여기에서는 참조 정보를 제공합니다.

문제 해결(작업)

이 장에는 Solaris 10 5/09 소프트웨어를 설치할 때 발생할 수 있는 구체적인 오류 메시지와 일반적인 문제 목록이 있습니다. 또한 그런 문제들에 대한 해결 방법도 설명합니다. 다음 절 목록을 사용하여 설치 프로세스에서 문제가 발생한 부분을 먼저 확인할 수 있습니다.

- 217 페이지 “네트워크 설치 설정 문제”
- 218 페이지 “시스템 부트 관련 문제”
- 224 페이지 “Solaris OS의 초기 설치”
- 226 페이지 “Solaris OS 업그레이드”

주 - "부트 가능 매체"라는 문구가 표시되면 Solaris 설치 프로그램과 JumpStart 설치 방법을 의미합니다.

네트워크 설치 설정 문제

Unknown client "*host_name*"

원인: `add_install_client` 명령 *host_name* 인수가 이름 지정 서비스의 호스트가 아닙니다.

해결책: 호스트 *host_name*을 이름 지정 서비스에 추가하고 `add_install_client` 명령을 다시 실행합니다.

오류: <system name> does not exist in the NIS ethers map

이를 추가하고 `add_install_client` 명령을 다시 실행함.

설명: `add_install_client` 명령을 실행하는 경우 위의 오류와 함께 명령이 실패합니다.

원인: 설치 서버에 추가 중인 클라이언트가 서버의 `/etc/ethers` 파일에 존재하지 않습니다.

해결책: 필요한 정보를 설치 서버의 /etc/ethers 파일에 추가하고 add_install_client 명령을 다시 실행합니다.

1. 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
2. 클라이언트에서 이더넷 주소를 찾습니다.

```
# ifconfig -a grep ethers
ether 8:0:20:b3:39:1d
```

3. 설치 서버에서 편집기를 사용하여 /etc/ethers 파일을 엽니다. 주소를 목록에 추가합니다.
4. 다음 예에서와 같이 클라이언트에서 add_install_client를 다시 실행합니다.

```
# ./add_install_client bluegill sun4u
```

시스템 부트 관련 문제

매체에서 부트할 때 나타나는 오류 메시지

le0: No carrier - transceiver cable problem

원인: 시스템이 네트워크에 연결되지 않았습니다.

해결책: 네트워크에 연결되지 않은 시스템인 경우 이 메시지를 무시합니다. 네트워크에 연결된 시스템인 경우 이더넷 케이블이 제대로 연결되어 있는지 확인합니다.

The file just loaded does not appear to be executable

원인: 시스템에서 부트를 위한 적합한 매체를 찾을 수 없습니다.

해결책: 시스템이 설치 서버에서 네트워크를 통해 Solaris 10 5/09 소프트웨어를 설치하는 데 적합하도록 설정되었는지 확인합니다. 확인할 내용의 예는 다음과 같습니다.

- Solaris Operating System DVD 또는 Solaris Software CD를 설치 서버에 복사하는 경우 시스템을 설정할 때 올바른 플랫폼 그룹을 지정했는지 확인하십시오.
- DVD 또는 CD 매체를 사용하는 경우 Solaris Operating System DVD 또는 Solaris Software - 1 CD 매체가 마운트되었으며 설치 서버에서 액세스할 수 있는지 확인합니다.

boot: cannot open <filename> (SPARC 기반 시스템에만 적용)

원인: 이 오류는 boot-file의 위치를 명시적으로 설정하여 위치를 무시할 때 발생합니다.

주 - `filename`은 영향이 미치는 이름의 변수입니다.

해결책: 다음 지침을 따릅니다.

- PROM에 있는 boot-파일을 “(공백)으로 재설정합니다.
- 진단 스위치가 꺼져 있고 참으로 설정되어 있는지 확인합니다.

Can't boot from file/device

원인: 설치 매체가 부트 가능한 매체를 찾을 수 없습니다.

해결책: 다음 조건이 충족되어 있는지 확인합니다.

- DVD-ROM 또는 CD-ROM 드라이브가 제대로 설치되어 있으며 켜져 있습니다.
- Solaris Operating System DVD 또는 Solaris Software - 1 CD가 드라이브에 들어 있습니다.
- 디스크가 손상되거나 먼지가 묻어있지 않습니다.

WARNING: clock gained xxx days -- CHECK AND RESET DATE! (x86 기반 시스템에만 적용)

설명: 이 메시지는 정보를 제공하는 메시지입니다.

해결책: 이 메시지를 무시하고 설치를 계속합니다.

Not a UFS file system(x86 기반 시스템에만 적용)

원인: Solaris 10 5/09 소프트웨어가 설치된 경우(Solaris 설치 프로그램 또는 사용자 정의 JumpStart 사용) 부트 디스크가 선택되지 않은 것입니다. 시스템을 부트하려면 지금 BIOS를 편집해야 합니다.

해결책: 부트할 BIOS를 선택합니다. 지침은 BIOS 설명서를 참조하십시오.

매체에서 부트할 때 발생하는 일반적인 문제

The system does not boot.

설명: 사용자 정의 JumpStart 서버를 처음 설정할 때 오류 메시지를 반환하지 않는 부트 문제가 발생할 수 있습니다. 시스템과 시스템 부트 방법에 관한 정보를 확인하려면 -v 옵션과 함께 부트 명령을 실행합니다. -v 옵션을 사용하는 경우 부트 명령이 화면에 대한 자세한 디버그 정보를 표시합니다.

주 - 이 플래그가 제공되지 않으면 메시지는 계속 인쇄되지만 출력은 시스템 로그 파일로 지정됩니다. 자세한 내용은 [syslogd\(1M\)](#)를 참조하십시오.

해결책: SPARC 기반 시스템의 경우 ok 프롬프트에서 다음 명령을 입력합니다.

ok boot net -v - install

Boot from DVD media fails on systems with Toshiba SD-M 1401 DVD-ROM

설명: 펌웨어 개정 1007인 Toshiba SD-M1401 DVD-ROM이 시스템에 설치되어 있는 경우 시스템이 Solaris Operating System DVD에서 부트할 수 없습니다.

해결책: 패치 111649-03 또는 이후 버전을 적용하여 Toshiba SD-M1401 DVD-ROM 드라이브의 펌웨어를 업데이트합니다. 패치 111649-03은 sunsolve.sun.com에서 사용 가능합니다.

메모리 카드가 아닌 PC 카드를 삽입하는 경우 시스템이 정지하거나 패닉 상태가 됩니다. (x86 기반 시스템에만 적용)

원인: 메모리가 아닌 PC 카드는 다른 장치가 사용하는 동일한 메모리 자원을 사용할 수 없습니다.

해결책: 이 문제를 수정하려면 PC 카드에 대한 지침을 참조하고 주소 범위를 확인합니다.

The system hangs before displaying the system prompt. (x86 기반 시스템에만 적용)

해결책: 지원되지 않는 하드웨어가 있습니다. 하드웨어 제조업체의 설명서를 확인하십시오.

네트워크에서 부트할 때 나타나는 오류 메시지

WARNING: getfile: RPC failed: error 5 (RPC Timed out).

설명: 이 오류는 설치 클라이언트의 부트 요청에 응답하는 서버가 네트워크상에 두 대 이상 있을 때 발생합니다. 설치 클라이언트가 잘못된 부트 서버에 연결하고 설치가 중단됩니다. 이 오류를 유발할 수 있는 구체적인 원인은 다음과 같습니다.

원인: 이유 1: /etc/bootparams 파일이 이 설치 클라이언트 항목이 있는 다른 서버에 있을 수 없습니다.

해결책: 이유 1: 네트워크 상의 서버에 설치 클라이언트용 /etc/bootparams 항목이 여러 개 있는지 확인합니다. 여러 항목이 있는 경우 설치 클라이언트에서 사용할 서버를 제외한 모든 설치 서버 및 부트 서버의 /etc/bootparams 파일에 있는 중복된 클라이언트 항목을 제거합니다.

원인: 이유 2: 설치 클라이언트용으로 여러 개의 /tftpboot 또는 /rplboot 디렉토리 입력 항목이 존재할 수 있습니다.

해결책: 이유 2: 네트워크 상의 서버에 설치 클라이언트용 /tftpboot 또는 /rplboot 디렉토리 항목이 여러 개 있는지 확인합니다. 여러 항목이 있는 경우 설치 클라이언트에서 사용할 서버를 제외한 모든 설치 서버와 부트 서버의 /tftpboot 또는 /rplboot 디렉토리에 있는 중복된 클라이언트 항목을 제거합니다.

원인: 이유 3: 서버의 /etc/bootparams 파일에 설치 클라이언트 항목이 있고 모든 시스템이 프로파일 서버에 액세스할 수 있는 또 다른 /etc/bootparams 파일에 항목이 있습니다. 이러한 항목은 다음과 유사합니다.

```
* install_config=profile_server:path
```

NIS 또는 NIS+ bootparams 테이블의 이전 항목과 유사한 행 역시 이런 오류를 유발할 수 있습니다.

해결책: 이유 3: 이름 지정 서비스 bootparams 맵 또는 테이블에 와일드카드 항목이 있으면(예: *install_config=) 해당 항목을 삭제하고 부트 서버의 /etc/bootparams 파일에 추가합니다.

No network boot server. Unable to install the system. See installation instructions. (x86 기반 시스템에만 적용)

원인: 이 오류는 네트워크에서 설치를 시도할 때 발생합니다. 시스템이 올바르게 설정되지 않았습니니다.

해결책: 네트워크에서 설치할 시스템을 올바르게 설정했는지 확인합니다. **Solaris 10 5/09 설치 설명서: 네트워크 기반 설치의 “CD 이미지로 네트워크에서 설치할 시스템 추가”**를 참조하십시오.

prom_panic: Could not mount file system(SPARC 기반 시스템에만 적용)

원인: 이 오류는 네트워크에서 Solaris를 설치하지만 부트 소프트웨어가 다음을 찾을 수 없는 경우 발생합니다.

- Solaris Operating System DVD, DVD 또는 설치 서버의 DVD 이미지 복사본
- Solaris Software - 1 CD 이미지, Solaris Software - 1 CD 또는 설치 서버의 CD 이미지 사본

해결책: 설치 소프트웨어가 마운트 및 공유되어 있는지 확인합니다.

- 설치 서버의 DVD-ROM 또는 CD-ROM 드라이브에서 Solaris를 설치하는 경우 Solaris Operating System DVD 또는 Solaris Software - 1 CD가 CD-ROM 드라이브에 삽입되었으며 마운트 되었는지, 또한 /etc/dfs/dfstab 파일에서 공유되는지 확인하십시오.
- 설치 서버 디스크의 Solaris Operating System DVD 이미지나 Solaris Software - 1 CD 이미지 복사본을 이용해 설치하려면 해당 복사본의 디렉토리 경로가 /etc/dfs/dfstab 파일에서 공유되어 있는지 확인합니다.

Timeout waiting for ARP/RARP packet... (SPARC 기반 시스템에만 적용)

원인: 이유 1: 클라이언트가 네트워크에서 부트를 시도하지만 해당 클라이언트를 인식하는 시스템이 없습니다.

해결책: 이유 1: 시스템의 호스트 이름이 NIS 또는 NIS+ 이름 지정 서비스에 있는지 확인합니다. 또한 부트 서버의 /etc/nsswitch.conf 파일에서 bootparams 검색 순서를 확인합니다.

예를 들어 /etc/nsswitch.conf 파일의 다음 행은 JumpStart 또는 Solaris 설치 프로그램이 먼저 NIS 맵에서 bootparams 정보를 찾는다라는 것을 나타냅니다. 프로그램이 정보를 찾지 못하면 설치 프로그램은 부트 서버의 /etc/bootparams 파일에서 찾습니다.

bootparams: nis files

원인: 이유 2: 클라이언트의 이더넷 주소가 올바르지 않습니다.

해결책: 이유 2: 설치 서버의 /etc/ethers 파일에 있는 클라이언트의 이더넷 주소가 올바른지 확인합니다.

원인: 이유 3: 사용자 JumpStart 설치에서 add_install_client 명령은 지정한 서버를 설치 서버로 사용하는 플랫폼 그룹을 지정합니다. add_install_client를 사용할 때 잘못된 구조 값이 사용되면 이 문제가 발생합니다. 예를 들어 설치하려는 컴퓨터는 sun4u이나 i86pc를 대신 사용했습니다.

해결책: 이유 3: 올바른 구조 값으로 add_install_client를 다시 실행합니다.

ip: joining multicasts failed on tr0 - will use link layer broadcasts for multicast(**x86 기반 시스템에만 적용**)

원인: 이 오류 메시지는 토큰 링 카드로 시스템을 부팅할 때 표시됩니다. 이더넷 멀티캐스트 및 토큰 링 멀티캐스트는 동일한 방법으로 작동하지 않습니다. 드라이버에서 잘못된 멀티캐스트 주소를 받았기 때문에 이 오류 메시지를 반환합니다.

해결책: 이 오류 메시지를 무시하십시오. 멀티캐스트가 작동하지 않으면 IP는 대신 계층 브로드캐스트를 사용하고 설치 실패를 유발하지 않습니다.

Requesting Internet address for *Ethernet_Address*(**x86 기반 시스템에만 적용**)

원인: 클라이언트가 네트워크에서 부트를 시도하지만 해당 클라이언트를 인식하는 시스템이 없습니다.

해결책: 시스템의 호스트 이름이 이름 지정 서비스에 나열되어 있는지 확인합니다. 시스템의 호스트 이름이 NIS 또는 NIS+ 이름 지정 서비스에 나열되어 있고 시스템이 이 오류 메시지를 계속 출력하면 재부팅해 보십시오.

RPC: Timed out No bootparams (whoami) server responding; still trying...(x86 기반 시스템에만 적용)

원인: 클라이언트가 네트워크에서 부트를 시도하고 있지만 설치 서버의 /etc/bootparams 파일에 항목이 있는 시스템을 찾을 수 없습니다.

해결책: 설치 서버에서 add_install_client를 사용합니다. 이 명령을 사용하면 /etc/bootparams 파일의 적절한 항목을 추가하므로 클라이언트가 네트워크에서 부트할 수 있습니다.

Still trying to find a RPL server...(x86 기반 시스템에만 적용)

원인: 시스템이 네트워크에서 부트를 시도하지만 서버가 이 시스템을 부트하도록 설정되지 않았습니다.

해결책: 설치 서버에서 시스템이 설치되도록 `add_install_client`를 실행합니다. `add_install_client` 명령이 필요한 네트워크 부트 프로그램을 포함하는 `/rplboot` 디렉토리를 설정합니다.

CLIENT MAC ADDR: FF FF FF FF FF FF(DHCP를 사용 네트워크 설치에만 적용)

원인: DHCP 서버가 제대로 구성되어 있지 않습니다. 옵션이나 매크로가 DHCP Manager 소프트웨어에 제대로 정의되어 있지 않은 경우 이 오류가 발생할 수 있습니다.

해결책: DHCP Manager 소프트웨어에서 옵션과 매크로가 제대로 정의되어 있는지 확인합니다. 라우터 옵션이 정의되어 있고 네트워크 설치에 사용할 서브넷에 대해 라우터 옵션 값이 올바른지 확인합니다.

네트워크에서 부트할 때 발생하는 일반적인 문제

시스템은 네트워크에서 부트하지만, 지정된 설치 서버 이외의 시스템에서 부트합니다.

원인: `/etc/bootparams` 및 `/etc/ethers` 항목이 클라이언트에 대한 다른 시스템에 있습니다.

해결책: 이름 서버에서 설치 중인 시스템에 대한 `/etc/bootparams` 항목을 업데이트합니다. 항목은 다음 구문과 일치해야 합니다.

```
install_system root=boot_server:path install=install_server:path
```

또한 `bootparams` 항목이 설치 클라이언트용 서브넷에 있는지 확인합니다.

The system does not boot from the network(DHCP 사용 네트워크 설치에만 적용).

원인: DHCP 서버가 제대로 구성되어 있지 않습니다. 시스템이 DHCP 서버에 설치 클라이언트로 구성되어 있지 않으면 이 오류가 발생할 수 있습니다.

해결책: DHCP 관리자 소프트웨어에서 클라이언트 시스템에 대해 설치 옵션과 매크로가 정의되어 있는지 확인합니다. 자세한 내용은 [Solaris 10 5/09 설치 설명서: 네트워크 기반 설치의 “DHCP 서비스를 사용하여 시스템 구성 정보 미리 구성\(작업\)”](#)을 참조하십시오.

Solaris OS의 초기 설치

Initial installation fails

해결책: Solaris 설치가 실패하면 설치를 다시 시작해야 합니다. 설치를 다시 시작하려면 Solaris Operating System DVD, Solaris Software - 1 CD 또는 네트워크에서 시스템을 부트합니다.

소프트웨어를 부분적으로 설치한 뒤 Solaris 소프트웨어를 설치 해제할 수 없습니다. 백업을 사용하여 시스템을 복원하거나 Solaris 설치 프로세스를 다시 시작해야 합니다.

/cdrom/cdrom0/SUNW xxxx/reloc.cpio: Broken pipe

설명: 이 오류 메시지는 정보를 제공하기 위한 것으로 설치에는 영향을 미치지 않습니다. 이 상태는 파이프에 대한 쓰기에 읽기 프로세스가 없는 경우 발생합니다.

해결책: 이 메시지를 무시하고 설치를 계속합니다.

WARNING: CHANGE DEFAULT BOOT DEVICE(x86 기반 시스템에만 적용)

원인: 이 메시지는 정보를 제공하는 메시지입니다. 시스템의 BIOS에 설정된 기본 부트 장치가 Solaris Device Configuration Assistant를 사용하여 시스템을 부트해야 하는 장치로 설정되었을 수 있습니다.

해결책: 설치를 계속하고, 필요한 경우 Solaris 소프트웨어를 설치한 후 BIOS에 지정된 시스템 기본 부트 장치를 Solaris Device Configuration Assistant가 필요하지 않은 장치로 변경합니다.

x86 전용 - locale 키워드를 사용하여 초기 설치에 대해 사용자 정의 JumpStart 프로필을 테스트하는 경우 pfinstall -D 명령을 사용하여 프로필을 테스트할 수 없습니다. 해결 방법은 226 페이지 “Solaris OS 업그레이드” 절의 “could not select locale” 오류 메시지를 참조하십시오.

▼ x86: IDE 디스크의 잘못된 블록 확인

IDE 디스크 드라이브는 Solaris 소프트웨어가 지원하는 다른 드라이브와 같이 잘못된 블록을 자동으로 매핑하지 않습니다. Solaris를 IDE 디스크에 설치하기 전에 디스크의 표면 분석을 수행하려고 할 수 있습니다. IDE 디스크에서 표면 분석을 수행하려면 다음 절차를 따릅니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 설치 매체로 부트합니다.
- 3 설치 유형을 선택하라는 프롬프트가 표시되면 옵션 6, 단일 사용자 셸을 선택합니다.
- 4 **format(1M)** 프로그램을 시작합니다.
format
- 5 표면을 분석을 수행하려는 IDE 디스크 드라이브를 지정합니다.
cx
cx 제어기 번호
dy 장치 번호
- 6 **fdisk** 분할 영역이 있는지 확인합니다.
 - Solaris **fdisk** 분할 영역이 이미 존재하는 경우 단계 7을 계속합니다.
 - Solaris **fdisk** 분할 영역이 존재하지 않으면 **fdisk** 명령을 사용하여 디스크에 Solaris 분할 영역을 만듭니다.

```
format> fdisk
```
- 7 표면을 분석을 시작하려면 다음을 입력합니다.
format> **analyze**
- 8 현재 설정을 결정하고 다음을 입력합니다.
analyze> **config**
- 9 (선택적) 설정을 변경하려면 다음을 입력합니다.
analyze> **setup**
- 10 잘못된 블록을 찾으려면 다음을 입력합니다.
analyze> *type_of_surface_analysis*
type_of_surface_analysis 읽기, 쓰기 또는 비교
format 도중 잘못된 블록이 있는 경우 다시 매핑합니다.
- 11 분석을 종료하려면 다음을 입력합니다.
analyze> **quit**
- 12 다시 매핑할 블록을 지정할 것인지 결정합니다.
 - 아니오일 경우 단계 13으로 이동합니다.
 - 지정하려면 다음을 입력합니다.

```
format> repair
```

- 13 포맷 프로그램을 종료하려면 다음을 입력합니다.

```
quit
```

- 14 다음 명령을 입력하여 다중 사용자 모드에서 매체를 다시 시작합니다.

```
# exit
```

Solaris OS 업그레이드

업그레이드할 때 나타나는 오류 메시지

No upgradable disks

원인: /etc/vfstab 파일에 있는 스왑 항목으로 인해 업그레이드가 실패하고 있습니다.

해결책: /etc/vfstab 파일에서 다음 행을 주석 처리합니다.

- 업그레이드되지 않는 디스크의 모든 스왑 파일 및 슬라이스
- 존재하지 않는 스왑 파일
- 사용하지 않는 스왑 슬라이스

usr/bin/bzcat not found

원인: 패치 클러스터가 없어 Solaris Live Upgrade가 실패하고 있습니다.

해결책: Solaris Live Upgrade 설치를 위한 패치가 필요합니다. <http://sunsolve.sun.com>을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 206844(이전의 72099)를 검색합니다.

Upgradeable Solaris root devices were found, however, no suitable partitions to hold the Solaris install software were found. Upgrading using the Solaris Installer is not possible. It might be possible to upgrade using the Solaris Software 1 CDROM. (x86 기반 시스템에만 적용)

원인: 공간이 부족하여 Solaris Software - 1 CD를 사용하여 업그레이드할 수 없습니다.

해결책: 업그레이드하려면 512 MB보다 크거나 같은 스왑 슬라이스를 만들거나 Solaris Operating System DVD의 Solaris 설치, 네트워크 설치 이미지 또는 JumpStart와 같은 다른 업그레이드 방법을 사용할 수 있습니다.

오류: Could not select locale(x86 기반 시스템에만 적용)

원인: pfinstall -D 명령을 사용하여 JumpStart 프로필을 테스트할 경우 다음과 같은 조건에서 연습(dry run) 테스트가 실패합니다.

- 프로필에 locale 키워드가 포함되어 있는 경우
- GRUB 소프트웨어가 포함된 릴리스를 테스트하는 경우 **Solaris 10 1/06** 릴리스부터 GRUB 부트로더에서 GRUB 메뉴를 사용하여 시스템에 설치한 다양한 운영 체제를 쉽게 부트할 수 있도록 지원합니다.

GRUB 소프트웨어의 소개로 미니루트가 압축됩니다. 소프트웨어는 압축된 미니루트에서 로컬 목록을 더 이상 찾을 수 없습니다. 미니루트는 최소 크기의 Solaris 루트(/) 파일 시스템이며 Solaris 설치 매체에 있습니다.

해결책: 다음 단계를 수행하십시오. 다음 값을 사용합니다.

- MEDIA_DIR은 /cdrom/cdrom0/
 - MINIROOT_DIR은 \$MEDIA_DIR /Solaris_10/Tools/Boot
 - MINIROOT_ARCHIVE는 \$MEDIA_DIR/boot/x86.miniroot
 - TEMP_FILE_NAME은 /tmp/test
1. 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.
 2. 미니루트 아카이브의 압축을 풉니다.

```
# /usr/bin/gzcat $MINIROOT_ARCHIVE > $TEMP_FILE_NAME
```
 3. lofiadm 명령을 사용하여 미니루트 장치를 만듭니다.

```
# LOFI_DEVICE=/usr/sbin/lofiadm -a $TEMP_FILE_NAME
# echo $LOFI_DEVICE
/dev/lofi/1
```
 4. Miniroot 디렉토리에서 lofi 명령을 사용하여 미니루트를 마운트합니다.

```
# /usr/sbin/mount -F ufs $LOFI_DEVICE $MINIROOT_DIR
```
 5. 프로필을 테스트합니다.

```
# /usr/sbin/install.d/pfinstall -D -c $MEDIA_DIR $path-to-jumpstart_profile
```
 6. 테스트가 완료되면 lofi 장치를 마운트 해제합니다.

```
# umount $LOFI_DEVICE
```
 7. lofi 장치를 삭제합니다.

```
# lofiadm -d $TEMP_FILE_NAME
```

업그레이드할 때 발생하는 일반적인 문제

The upgrade option is not presented even though there is a version of Solaris software that's upgradable on the system.

원인: 이유 1: /var/sadm 디렉토리가 symlink이거나 다른 파일 시스템으로부터 마운트되었습니다.

해결책: 이유 1: /var/sadm 디렉토리를 루트(/) 또는 /var 파일 시스템으로 이동합니다.

원인: 이유 2: /var/sadm/softinfo/INST_RELEASE 파일이 없습니다.

해결책: 이유 2: 다음 템플릿을 사용하여 새로운 INST_RELEASE 파일을 만듭니다.

```
OS=Solaris
VERSION=x
REV=0
```

x 시스템상의 Solaris 소프트웨어 버전입니다.

원인: 이유 3: SUNWusr이 /var/sadm/softinfo에 없습니다.

해결책: 해결 방법 3: 초기 설치를 수행해야 합니다. Solaris 소프트웨어를 업그레이드할 수 없습니다.

Couldn't shut down or initialize the md driver

해결책: 다음 지침을 따릅니다.

- 파일 시스템이 RAID-1 볼륨이 아닌 경우 vsftab 파일에서 주석 처리합니다.
- 파일 시스템이 RAID-1 볼륨인 경우 미러를 해체하고 다시 설치합니다. 미러 해체에 대한 자세한 내용은 [Solaris Volume Manager Administration Guide](#)의 “Removing RAID-1 Volumes (Unmirroring)”를 참조하십시오.

The upgrade fails because the Solaris installation program cannot mount a file system.

원인: 업그레이드 도중 스크립트는 시스템의 /etc/vfstab 파일에 나열된 모든 파일 시스템을 업그레이드 중인 루트(/) 파일 시스템에 마운트하려고 시도합니다. 설치 스크립트가 파일 시스템을 마운트할 수 없으면, 업그레이드가 실패하여 종료됩니다.

해결책: 시스템의 /etc/vfstab 파일 내의 모든 파일 시스템이 마운트될 수 있는지 확인하십시오. 마운트할 수 없거나 문제를 야기하는 /etc/vfstab 파일의 파일 시스템을 주석 처리하여 Solaris 설치 프로그램이 업그레이드하는 동안에 해당 시스템에 대한 마운트를 시도하지 않도록 합니다. 업그레이드할 소프트웨어(예: /usr)가 포함된 모든 시스템 기반 파일 시스템은 주석으로 처리할 수 없습니다.

The upgrade fails

설명: 시스템에 업그레이드를 위한 충분한 공간이 없습니다.

원인: **Solaris 10 5/09 설치 설명서: 설치 및 업그레이드 계획의 “디스크 공간 재할당을 통해 업그레이드”**에서 공간 문제를 확인하고 자동 레이아웃을 사용하여 공간을 재할당하지 않고 문제를 해결할 수 있는지 확인합니다.

RAID-1 볼륨 루트 (/) 파일 시스템 업그레이드 문제

해결책: 루트 (/) 파일 시스템인 Solaris Volume Manager RAID-1 볼륨을 사용할 때 업그레이드 문제가 발생하는 경우 **Solaris Volume Manager Administration Guide**의 25 장, “**Troubleshooting Solaris Volume Manager (Tasks)**”를 참조하십시오.

▼ 업그레이드 실패 후에 업그레이드 계속 진행

업그레이드가 실패하고 시스템을 소프트 부트할 수 없습니다. 정전이나 네트워크 연결 오류와 같이 사용자가 해결할 수 있는 수준의 오류가 아닙니다.

- 1 **Solaris Operating System DVD, Solaris Software - 1 CD** 또는 네트워크에서 시스템을 다시 부트합니다.
- 2 설치를 위한 업그레이드 옵션을 선택합니다.
시스템이 부분적으로 업그레이드되었는지 여부를 Solaris 설치 프로그램이 확인하고 업그레이드를 계속합니다.

x86: GRUB 사용 시 Solaris Live Upgrade 문제

x86 기반 시스템에서 Solaris Live Upgrade 및 GRUB 부트 로더를 사용할 경우 다음과 같은 오류가 발생할 수 있습니다.

오류: The media product tools installation directory *path-to-installation-directory* does not exist.

오류: The media *dirctory* does not contain an operating system upgrade image.

설명: 이러한 오류 메시지는 `luupgrade` 명령을 사용하여 새 부트 환경을 업그레이드할 때 표시됩니다.

원인: 이전 버전의 Solaris Live Upgrade를 사용하고 있습니다. 시스템에 설치한 Solaris Live Upgrade 패키지가 매체 및 해당 매체의 릴리스와 호환되지 않습니다.

해결책: 항상 업그레이드하는 릴리스의 Solaris Live Upgrade 패키지를 사용합니다.

예: 다음 예에서 오류 메시지는 시스템의 Solaris Live Upgrade 패키지와 매체에 있는 패키지의 버전이 같지 않음을 나타냅니다.

```
# luupgrade -u -n s10u1 -s /mnt
Validating the contents of the media </mnt>.
The media is a standard Solaris media.
ERROR: The media product tools installation directory
</mnt/Solaris_10/Tools/Boot/usr/sbin/install.d/install_config> does
not exist.
ERROR: The media </mnt> does not contain an operating system upgrade
image.
```

오류: Cannot find or is not executable: </sbin/biosdev>.

오류: One or more patches required by Solaris Live Upgrade has not been installed.

원인: Solaris Live Upgrade에 필요한 패치가 하나 이상 시스템에 설치되어 있지 않습니다. 이 오류 메시지는 누락된 모든 패치를 나타내지는 않습니다.

해결책: Solaris Live Upgrade를 사용하기 전에 항상 필요한 패치를 모두 설치하십시오. <http://sunsolve.sun.com>을 참조하여 가장 최근에 업데이트된 패치 목록이 있는지 확인합니다. SunSolve 웹 사이트에서 정보 문서 206844(이전의 72099)를 검색합니다.

오류: Device mapping command </sbin/biosdev> failed. Please reboot and try again.

원인: 이유 1: 이전 관리 작업 때문에 Solaris Live Upgrade에서 장치를 매핑할 수 없습니다.

해결책: 이유 1: 시스템을 재부트한 후 Solaris Live Upgrade를 다시 시도하십시오.

원인: 이유 2: 시스템을 재부트한 후에도 동일한 오류 메시지가 표시되면 동일한 디스크가 두 개 이상 있습니다. 장치 매핑 명령이 디스크를 구분할 수 없습니다.

해결책: 이유 2: 디스크 중 하나에 새 임식 fdisk 분할 영역을 만듭니다. [fdisk\(1M\)](#) 매뉴얼 페이지를 참조하십시오. 그런 다음 시스템을 재부트합니다.

Cannot delete the boot environment that contains the GRUB menu

원인: Solaris Live Upgrade는 부트 환경에 GRUB 메뉴가 있을 경우 부트 환경을 삭제할 수 없도록 제한합니다.

해결책: 이 부트 환경을 다시 사용하려면 [lumake\(1M\)](#) 또는 [luupgrade\(1M\)](#) 명령을 사용합니다.

The file system containing the GRUB menu was accidentally remade. However, the disk has the same slices as before. For example, the disk was not re-sliced.

원인: GRUB 메뉴가 포함된 파일 시스템은 시스템을 부팅 가능하도록 유지하는 데 중요합니다. Solaris Live Upgrade 명령은 GRUB 메뉴를 삭제하지 않습니다. 그러나 Solaris Live Upgrade 명령이 아닌 다른 명령으로 GRUB 메뉴가 포함된 파일 시스템을 실수로 다시 만들거나 삭제하면 복구 소프트웨어가 GRUB 메뉴를 다시 설치합니다.

복구 소프트웨어는 다음에 재부트할 때 GRUB 메뉴를 동일한 파일 시스템에 다시 배치합니다. 예를 들어, 파일 시스템에서 `newfs` 또는 `mkfs` 명령을 사용하여 실수로 GRUB 메뉴를 삭제했을 수 있습니다. GRUB 메뉴를 올바르게 복원하려면 슬라이스가 다음 조건을 만족해야 합니다.

- 마운트 가능 파일 시스템 포함
- 슬라이스가 이전에 위치해 있던 Solaris Live Upgrade 부트 환경의 일부 유지

시스템을 재부트하기 전에 슬라이스에서 필요한 수정 작업을 수행합니다.

해결책: 시스템을 다시 부트합니다. GRUB 메뉴의 백업 복사본이 자동으로 설치됩니다.

GRUB 메뉴의 `menu.lst` 파일이 실수로 삭제되었습니다.

해결책: 시스템을 다시 부트합니다. GRUB 메뉴의 백업 복사본이 자동으로 설치됩니다.

▼ Veritas VxVm을 실행하는 Solaris Live Upgrade를 사용하여 업그레이드하는 중에 시스템 장애 발생

실행 중인 Veritas VxVM을 업그레이드하는 동안 Solaris Live Upgrade를 사용하는 경우 다음 방법으로 업그레이드하지 않는 한 다시 부트할 때 시스템에 장애가 발생합니다. 패키지가 Solaris 고급 패키징 지침 사항을 따르지 않으면 문제가 발생합니다.

- 1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 비활성 부트 환경을 만듭니다. [62 페이지 “새 부트 환경 생성”](#)를 참조하십시오.
- 3 비활성 부트 환경을 업그레이드하기 전에 비활성 부트 환경에서 기존 Veritas 소프트웨어를 비활성화시켜야 합니다.

- a. 비활성 부트 환경을 마운트합니다.

```
# lumount inactive_boot_environment_name mount_point
```

예를 들면 다음과 같습니다.

```
# lumount solaris8 /mnt
```

- b. `vfstab`가 포함된 디렉토리로 변경합니다. 예를 들면 다음과 같습니다.

```
# cd /mnt/etc
```

- c. 비활성 부트 환경의 `vfstab` 파일의 사본을 만듭니다. 예를 들면 다음과 같습니다.

```
# cp vfstab vfstab.501
```

- d. 복사한 `vfstab`에서 Veritas 파일 시스템 항목을 모두 주석 처리합니다. 예를 들면 다음과 같습니다.

```
# sed '/vx\dsk/s/^\#/g' < vfstab > vfstab.novxfs
```

각 행의 첫 번째 문자는 해당 행을 주석 행으로 만드는 #으로 변경됩니다. 참고로 이 주석 행은 시스템 파일 주석 행과 다릅니다.

- e. 변경된 `vfstab` 파일을 복사합니다. 예를 들면 다음과 같습니다.

```
# cp vfstab.novxfs vfstab
```

- f. 비활성 부트 환경의 시스템 파일로 디렉토리를 변경합니다. 예를 들면 다음과 같습니다.

```
# cd /mnt/etc
```

- g. 비활성 부트 환경의 시스템 파일 사본을 만듭니다. 예를 들면 다음과 같습니다.

```
# cp system system.501
```

- h. `drv/vx`를 포함하는 모든 "forceload:" 항목을 주석 처리합니다.

```
# sed '/forceload: drv\/vx/s/^\*/' <system> system.novxfs
```

각 행의 첫 번째 문자는 해당 행을 명령줄로 만드는 *로 변경됩니다. 이 주석 행은 `vfstab` 파일 주석 행과 다르다는 점에 주의하십시오.

- i. Veritas `install-db` 파일을 만듭니다. 예를 들면 다음과 같습니다.

```
# touch vx/reconfig.d/state.d/install-db
```

- j. 비활성 부트 환경을 마운트 해제합니다.

```
# luumount inactive_boot_environment_name
```

- 4 비활성 부트 환경을 업그레이드합니다. 5 장, "Solaris Live Upgrade를 통해 업그레이드(작업)"를 참조하십시오.
- 5 비활성 부트 환경을 활성화시킵니다. 107 페이지 "부트 환경 활성화"를 참조하십시오.
- 6 시스템을 종료합니다.

```
# init 0
```

- 7 단일 사용자 모드에서 비활성 부트 환경을 부트합니다.

```
OK boot -s
```

“vxvm” 또는 “VXVM”을 포함하는 몇 개의 메시지와 오류 메시지는 무시할 수 있습니다. 비활성 부트 환경이 활성화됩니다.

8 Veritas를 업그레이드합니다.

a. 시스템에서 Veritas VRTSvmsa 패키지를 제거합니다. 예:

```
# pkgrm VRTSvmsa
```

b. 디렉토리를 Veritas 패키지로 변경합니다.

```
# cd /location_of_Veritas_software
```

c. 시스템에 최신 Veritas 패키지를 추가합니다.

```
# pkgadd -d 'pwd' VRTSvxvm VRTSvmsa VRTSvmdoc VRTSvman VRTSvmdv
```

9 원래의 vfstab 및 시스템 파일을 복원합니다.

```
# cp /etc/vfstab.original /etc/vfstab
```

```
# cp /etc/system.original /etc/system
```

10 시스템을 다시 부트합니다.

```
# init 6
```

x86: 기존 Service 분할 영역이 없는 시스템에서 기본적으로 만들지 않는 Service 분할 영역

현재 서비스 또는 진단 분할 영역이 없는 시스템에 Solaris 10 5/09 OS를 설치하는 경우 설치 프로그램에서 기본적으로 서비스 분할 영역을 만들지 못할 수 있습니다. Solaris 분할 영역과 동일한 디스크에 서비스 분할 영역을 포함하려는 경우 Solaris 10 5/09 OS를 설치하기 전에 서비스 분할 영역을 다시 작성해야 합니다.

서비스 분할 영역이 있는 시스템에 Solaris 8 2/02 OS를 설치했으면 설치 프로그램이 설치 분할 영역을 보존하지 않았을 수 있습니다. fdisk 부트 분할 영역을 직접 편집하여 서비스 분할 영역을 보존하지 않은 경우 설치 프로그램은 설치 중에 서비스 분할 영역을 삭제합니다.

주 - Solaris 8 2/02 OS를 설치할 때 서비스 분할 영역을 별도로 보존하지 않은 경우 서비스 분할 영역을 다시 작성할 수 없으며 Solaris 10 5/09 OS로 업그레이드하지 못할 수 있습니다.

Solaris 분할 영역이 있는 디스크에 서비스 분할 영역을 포함하려면 다음 해결 방법 중 한 가지를 선택합니다.

▼ 네트워크 설치 이미지 또는 Solaris Operating System DVD에서 소프트웨어 설치

네트워크 설치 이미지 또는 네트워크를 통해 Solaris Operating System DVD에서 소프트웨어를 설치하려면 다음 단계를 따르십시오.

- 1 디스크의 내용을 삭제합니다.
- 2 설치하기 전에 시스템용 진단 CD를 사용하여 서비스 분할 영역을 만듭니다.
서비스 분할 영역을 만드는 방법은 하드웨어 설명서를 참조하십시오.
- 3 네트워크에서 시스템을 부트합니다.
Customize fdisk Partitions 화면이 표시됩니다.
- 4 기본 부트 디스크 분할 영역 레이아웃을 로드하려면 기본값을 누릅니다.
설치 프로그램이 서비스 분할 영역을 보존하고 Solaris 분할 영역을 만듭니다.

▼ Solaris Software - 1 CD 또는 네트워크 설치 이미지에서 설치

Solaris 설치 프로그램을 사용하여 Solaris Software - 1 CD 또는 부트 서버에 있는 네트워크 설치 이미지에서 설치하려면 다음과 같이 합니다.

- 1 디스크의 내용을 삭제합니다.
- 2 설치하기 전에 시스템용 진단 CD를 사용하여 서비스 분할 영역을 만듭니다.
서비스 분할 영역을 만드는 방법은 하드웨어 설명서를 참조하십시오.
- 3 설치 프로그램이 Solaris 분할 영역을 만드는 방법을 선택하라는 메시지를 표시합니다.
- 4 시스템을 부트합니다.
- 5 Use rest of disk for Solaris partition 옵션을 선택합니다.
설치 프로그램이 서비스 분할 영역을 보존하고 Solaris 분할 영역을 만듭니다.
- 6 설치를 완료합니다.

추가 SVR4 패키지 요구 사항(참조)

이 부록은 패키지, 특히 타사 패키지를 설치하거나 제거하는 시스템 관리자용입니다. 이 패키지 요구 사항을 준수하여 다음 작업을 수행할 수 있습니다.

- Solaris Live Upgrade를 사용하여 업그레이드하고 비전역 영역과 디스크 없는 클라이언트 생성 및 유지를 할 수 있도록 현재 실행 중인 시스템이 수정되지 않도록 방지
- 사용자 정의 JumpStart 등의 설치 프로그램을 사용하여 설치를 자동화할 때 패키지가 대화식이 되지 않도록 방지

이 장은 다음 절로 구성되어 있습니다.

- 235 페이지 “현재 OS 수정 방지”.
- 239 페이지 “설치 또는 업그레이드 중 사용자 상호 작용 방지”.
- 240 페이지 “영역의 패키지 매개 변수 설정”

현재 OS 수정 방지

여기의 요구 사항을 준수하면 현재 OS가 변경되지 않도록 유지할 수 있습니다.

절대 경로 사용

운영 체제 설치가 성공하려면 패키지가 Solaris Live Upgrade 비활성 부트 환경 등의 대체 루트(/) 파일 시스템을 인지하고 올바르게 처리해야 합니다.

패키지는 pkgmap 파일(패키지 맵)에 절대 경로를 포함할 수 있습니다. 이 파일이 존재할 경우 해당 파일은 pkgadd 명령의 -R 옵션에 상대적으로 작성됩니다. 절대 및 상대(위치 변경이 가능한) 경로를 모두 포함하는 패키지를 대체 루트(/) 파일 시스템에 설치할 수도 있습니다. \$PKG_INSTALL_ROOT는 절대 및 재할당 가능한 파일에 모두 추가할 수 있으므로 pkgadd를 사용하여 설치할 때 모든 경로가 적절히 변환됩니다.

pkgadd -R 명령 사용

pkgadd -R 옵션을 사용하여 설치되거나 pkgrm -R 옵션을 사용하여 제거되는 패키지는 현재 실행 중인 시스템을 변경하지 말아야 합니다. 이 기능은 사용자 정의 JumpStart, Solaris Live Upgrade, 비전역 영역 및 디스크 없는 클라이언트에서 사용됩니다.

pkgadd 명령 -R 옵션을 사용하여 설치되거나 pkgrm 명령 -R 옵션을 사용하여 제거되는 패키지에 포함된 모든 프로시저 스크립트는 현재 실행 중인 시스템을 변경하지 말아야 합니다. 사용자가 입력하는 모든 설치 스크립트는 반드시 접두어가 \$PKG_INSTALL_ROOT 변수인 모든 디렉토리 또는 파일을 참조해야 합니다. 패키지는 \$PKG_INSTALL_ROOT 접두어가 있는 모든 디렉토리 및 파일을 작성해야 합니다. 패키지는 \$PKG_INSTALL_ROOT 접두어가 없는 디렉토리를 제거하면 안 됩니다.

표 B-1에서 스크립트 구문 예를 제공합니다.

표 B-1 설치 스크립트 구문의 예

스크립트 유형	올바른 구문	잘못된 구문
Bourne 셸 "if" 명령문 조각	<pre>if [-f \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf] ; then</pre>	<pre>if [-f /etc/myproduct.conf] ; \ then</pre>
파일 제거	<pre>/bin/rm -f \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf</pre>	<pre>/bin/rm -f /etc/myproduct.conf</pre>
파일 변경	<pre>echo "test=no" > \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf</pre>	<pre>echo "test=no" > \ /etc/myproduct.conf</pre>

\$PKG_INSTALL_ROOT 및 \$BASEDIR 사이의 차이점 개요

\$PKG_INSTALL_ROOT는 패키지를 추가할 컴퓨터의 루트(/) 파일 시스템의 위치입니다. 위치는 pkgadd 명령의 -R 인자로 설정됩니다. 예를 들어, 다음 명령이 호출되면 \$PKG_INSTALL_ROOT가 패키지를 설치하는 동안 /a가 됩니다.

```
# pkgadd -R /a SUNWvxxm
```

\$BASEDIR은 위치 변경이 가능한 패키지 객체가 설치된 위치 변경이 가능한 기본 디렉토리를 가리킵니다. 위치 변경이 가능한 객체만 여기에 설치됩니다. 위치 변경이 불가능한 객체(pkgmap 파일의 절대 경로가 있는 객체)는 항상 비활성 부트 환경에 상대적으로 설치되지만 실제로 \$BASEDIR에 상대적으로 설치되지는 않습니다. 패키지에 위치를 변경할 수 있는 객체가 없는 경우 이 패키지는 절대 패키지(위치 변경 불가능)라고 하며, \$BASEDIR가 정의되지 않고 패키지 프로시저 스크립트에서 사용할 수 없습니다.

예를 들어, 패키지의 pkgmap 파일은 다음 두 항목을 가지는 것으로 가정합니다.

```
1 f none sbin/ls 0555 root sys 3541 12322 1002918510
1 f none /sbin/ls2 0555 root sys 3541 12322 2342423332
```

pkginfo 파일에는 \$BASEDIR 의 다음 사양이 있습니다.

```
BASEDIR=/opt
```

이 패키지가 다음 명령으로 설치되면 ls는 /a/opt/sbin/ls에 설치되지만 ls2는 /a/sbin/ls2에 설치됩니다.

```
# pkgadd -R /a SUNWtst
```

스크립트 작성 지침

패키지 프로시저 스크립트는 반드시 현재 실행 중인 OS와 독립적이어서 OS가 수정되지 않도록 해야 합니다. 프로시저 스크립트는 패키지를 설치 및 제거하는 동안 특정 지점에서 발생하는 작업을 정의합니다. 이러한 미리 지정된 이름으로 만들 수 있는 프로시저 스크립트는 preinstall, postinstall, preremove, postremove 등 4개입니다.

표 B-2 스크립트 작성 지침

지침	Solaris Live Upgrade에 미치는 영향	비전역 영역에 미치는 영향
스크립트는 반드시 Bourne 셸(/bin/sh)에서 작성되어야 합니다. Bourne 셸은 pkgadd 명령이 프로시저 스크립트를 실행하는 데 사용하는 해석기입니다.	X	X
스크립트는 프로세스를 시작 또는 정지하면 안 되며, 운영 체제에 종속적이고 현재 실행 중인 시스템에 대한 정보를 보고하는 ps 또는 truss와 같은 명령의 출력에 종속되어서도 안 됩니다.	X	X
스크립트는 expr, cp, ls 및 셸 스크립트 작성을 용이하게 하는 기타 명령과 같은 기타 표준 UNIX 명령을 자유롭게 사용합니다.	X	X
스크립트가 호출하는 모든 명령은 패키지가 해당하는 모든 릴리스에서 실행해야 하기 때문에 지원되는 모든 릴리스에서 사용 가능해야 합니다. 따라서 Solaris 8 릴리스 이후에 추가 또는 제거된 명령은 사용할 수 없습니다.	X	
특정 명령 또는 옵션이 Solaris 8, 9 또는 10 릴리스에서 지원되는지 확인하려면 http://docs.sun.com 에서 Solaris Reference Manual AnswerBook의 해당 버전을 참조하십시오.		

디스크 없는 클라이언트 호환성 관리

패키지가 패키지 자체가 제공하는 명령을 실행하면 안 됩니다. 이는 디스크 없는 클라이언트 호환성을 유지하고 아직 설치되지 않은 공유 라이브러리가 필요할 수 있는 명령 실행을 방지하기 위한 것입니다.

패키지 검증

모든 패키지는 `pkgchk` 검증을 통과해야 합니다. 패키지가 작성되면 설치되기 전에 다음 명령으로 검사해야 합니다.

```
# pkgchk -d dir_name pkg_name
```

`dir_name` 패키지가 상주하는 디렉토리의 이름을 지정합니다.

`pkg_name` 패키지의 이름을 지정합니다.

예 B-1 패키지 테스트

패키지가 작성된 후 `pkgadd`에 대한 `-R dir_name` 옵션으로 대체 루트(/) 파일 시스템 위치에 설치하여 테스트해야 합니다. 패키지가 설치된 후 반드시 이 예와 같이 `pkgchk` 명령을 사용하여 올바른지 확인해야 합니다.

```
# pkgadd -d . -R /a SUNWvxvm
```

```
# pkgchk -R /a SUNWvxvm
```

오류가 표시되어서는 안 됩니다.

예 B-2 /export/SUNWvxvm에서 패키지 테스트

패키지가 /export/SUNWvxvm에 있는 경우 다음 명령을 실행합니다.

```
# pkgchk -d /export SUNWvxvm
```

오류가 표시되어서는 안 됩니다.

파일을 생성, 수정 및 삭제할 때 다른 명령으로 패키지를 확인할 수 있습니다. 다음 명령을 예로 들 수 있습니다.

- 예를 들어, `dircmp` 또는 `fssnap` 명령은 패키지가 제대로 작동하는지 검증하는 데 사용할 수 있습니다.
- 또한 `ps` 명령은 데몬이 패키지에 의해 정지 또는 시작되지 않았음을 확인하여 데몬 적합성을 테스트하기 위해 사용될 수 있습니다.
- `truss`, `pkgadd -v` 및 `pkgrm` 명령은 런타임 패키지 설치 적합성을 테스트할 수 있지만 일부 상황에서 작동하지 않을 수 있습니다. 다음 예에서, `truss` 명령은 모든 읽기 전용, 비`$TMPDIR` 액세스를 제거하고 지정된 비활성 부트 환경 내에 있지 않는 경로에 대한 비읽기 전용 액세스만 보여 줍니다.

```
# TMPDIR=/a; export TMPDIR
```

```
# truss -t open /usr/sbin/pkgadd -R ${TMPDIR} SUNWvxvm \
```

```
2>&1 > /dev/null | grep -v O_RDONLY | grep -v \
'open('"'${TEMPDIR}
```

설치 또는 업그레이드 중 사용자 상호 작용 방지

다음 표준 Solaris 유틸리티를 사용하는 경우 반드시 사용자에게 정보를 요구하는 프롬프트 없이 패키지가 추가 또는 제거되어야 합니다.

- 사용자 정의 JumpStart 프로그램
- Solaris Live Upgrade
- Solaris 설치 프로그램
- Solaris 영역

패키지가 사용자 상호 작용 없이 설치되는지 테스트하려면 새 관리 파일을 `pkgadd command -a` 옵션과 함께 설정합니다. `-a` 옵션은 기본 관리 파일 대신 사용될 설치 관리 파일을 정의합니다. 기본 파일을 사용하면 사용자에게 추가 정보를 묻는 메시지가 표시될 수 있습니다. 이들 검사를 무시하고 사용자 확인이 없이 패키지를 설치해야 함을 `pkgadd`에게 알려주는 관리 파일을 만들 수 있습니다. 자세한 내용은 매뉴얼 페이지 `admin(4)` 또는 `pkgadd(1M)`를 참조하십시오.

`pkgadd` 명령이 관리 파일을 사용하는 방식은 다음 예와 같습니다.

- 관리 파일이 제공되지 않으면 `pkgadd`는 `/var/sadm/install/admin/default`를 사용합니다. 이 파일을 사용하면 사용자 상호 작용이 생길 수 있습니다.

```
# pkgadd
```

- 상대 관리 파일이 명령줄에 제공되면 `pkgadd`는 `/var/sadm/install/admin`에서 파일 이름을 찾아 사용합니다. 이 예에서 상대 관리 파일의 이름은 `nocheck`이고 `pkgadd`는 `/var/sadm/install/admin/nocheck`를 찾습니다.

```
# pkgadd -a nocheck
```

- 절대 파일이 제공되면 `pkgadd`가 해당 파일을 사용합니다. 이 예에서 `pkgadd`는 `/tmp`에서 `nocheck` 관리 파일을 찾습니다.

```
# pkgadd -a /tmp/nocheck
```

예 B-3 설치 관리 파일

다음은 `pkgadd` 유틸리티와의 사용자 상호 작용이 거의 필요 없는 설치 관리 파일의 예입니다. 패키지가 시스템에서 사용 가능한 공간보다 더 많은 공간을 필요로 하는 경우를 제외하고 `pkgadd` 유틸리티는 이 파일을 사용하고 사용자에게 추가 정보를 묻는 메시지를 표시하지 않고 패키지를 설치합니다.

예 B-3 설치 관리 파일 (계속)

```
mail=  
instance=overwrite  
partial=nocheck  
runlevel=nocheck  
idepend=nocheck  
space=ask  
setuid=nocheck  
conflict=nocheck  
action=nocheck  
basedir=default
```

영역의 패키지 매개 변수 설정

패키지에는 비전역 영역이 설치된 시스템에서 내용을 배포하여 표시하는 방법을 제어하는 매개 변수가 있습니다. SUNW_PKG_ALLZONES, SUNW_PKG_HOLLOW 및 SUNW_PKG_THISZONE 패키지 매개 변수는 영역이 설치된 시스템에서 패키지의 특성을 정의합니다. 비전역 영역이 있는 시스템에서 패키지를 관리하려면 이러한 매개 변수를 설정해야 합니다.

다음 표에는 패키지 매개 변수 설정을 위한 네 가지 유효한 조합이 나열되어 있습니다. 다음 표에 나열되지 않은 조합을 설정할 경우 해당 조합은 유효하지 않으므로 패키지가 설치되지 않습니다.

주 - 세 개의 패키지 매개 변수를 모두 설정했는지 확인합니다. 세 개의 패키지 매개 변수를 모두 공백으로 둘 수 있습니다. 패키지 도구는 누락된 영역 패키지 매개 변수를 설정값이 “false”인 것처럼 해석하지만, 이 매개 변수는 설정하는 것이 좋습니다. 세 개의 패키지 매개 변수를 모두 설정하여 패키지를 설치하거나 제거할 때 패키지 도구가 나타내는 동작을 정확하게 지정합니다.

표 B-3 영역에 대해 유효한 패키지 매개 변수 설정

SUNW_PKG_ALLZONES 설정	SUNW_PKG_HOLLOW 설정	SUNW_PKG_THISZONE 설정	패키지 설명
false	false	false	<p>모든 영역 패키지 매개 변수에 대한 값을 지정하지 않은 패키지의 기본 설정입니다.</p> <p>이러한 설정을 가진 패키지는 전역 영역이나 비전역 영역에 설치할 수 있습니다.</p> <ul style="list-style-type: none"> ■ pkgadd 명령을 전역 영역에서 실행할 경우 패키지는 전역 영역과 모든 비전역 영역에 설치됩니다. ■ pkgadd 명령을 비전역 영역에서 실행할 경우 패키지는 비전역 영역에만 설치됩니다. <p>두 경우 모두 패키지가 설치된 모든 영역에 패키지의 전체 내용이 표시됩니다.</p>
false	false	true	<p>이러한 설정을 가진 패키지는 전역 영역이나 비전역 영역에 설치할 수 있습니다. 설치 후에 새로운 비전역 영역을 만든 경우 패키지는 새로운 비전역 영역에 전파되지 않습니다.</p> <ul style="list-style-type: none"> ■ pkgadd 명령을 전역 영역에서 실행할 경우 패키지는 전역 영역에만 설치됩니다. ■ pkgadd 명령을 비전역 영역에서 실행할 경우 패키지는 비전역 영역에만 설치됩니다. <p>두 경우 모두 패키지가 설치된 영역에 패키지의 전체 내용이 표시됩니다.</p>
true	false	false	<p>이러한 설정을 가진 패키지는 전역 영역에만 설치할 수 있습니다. pkgadd 명령을 실행하면 패키지가 전역 영역과 모든 비전역 영역에 설치됩니다. 모든 영역에 패키지의 전체 내용이 표시됩니다.</p> <p>주 - 이 패키지는 비전역 영역에 설치되지 않습니다.</p>

표 B-3 영역에 대해 유효한 패키지 매개 변수 설정 (계속)

SUNW_PKG_ALLZONES 설정	SUNW_PKG_HOLLOW 설정	SUNW_PKG_THISZONE 설정	패키지 설명
true	true	false	<p>이러한 설정을 가진 패키지는 전역 관리자가 전역 영역에만 설치할 수 있습니다. pkgadd 명령을 실행하면 패키지의 내용이 전역 영역에 전체 설치됩니다. 패키지 매개 변수가 이러한 값으로 설정된 경우 패키지 내용이 비전역 영역에 제공되지 않습니다. 설치할 패키지를 나타내는 데 필요한 패키지 설치 정보만 모든 비전역 영역에 설치됩니다. 그러면 이 패키지에 종속되는 다른 패키지를 설치할 수 있습니다. “빈” 패키지에 대한 자세한 내용은 System Administration Guide: Solaris Containers-Resource Management and Solaris Zones의 24 장, “About Packages and Patches on a Solaris System With Zones Installed (Overview)”를 참조하십시오.</p> <p>패키지 종속성 검사를 위해 패키지가 모든 영역에서 설치된 것으로 나타납니다.</p> <ul style="list-style-type: none"> ■ 전역 영역에는 패키지의 전체 내용이 표시됩니다. ■ 전체 루트 비전역 영역에는 패키지의 전체 내용이 표시되지 않습니다. ■ 비전역 영역이 전역 영역에서 파일 시스템을 상속하는 경우 이 파일 시스템에 설치된 패키지가 비전역 영역에 표시됩니다. 패키지가 제공하는 모든 다른 파일은 비전역 영역에 표시되지 않습니다. <p>예를 들어, 희소 루트 비전역 영역은 특정 디렉토리를 전역 영역과 공유합니다. 이러한 디렉토리는 읽기 전용입니다. 희소 루트 비전역 영역은 /platform 파일 시스템을 서로 공유합니다. 다른 예로는 부트 하드웨어에만 관련된 파일을 제공하는 패키지가 있습니다.</p> <p>주 - 이 패키지는 비전역 영역에 설치되지 않습니다.</p>
설명	자세한 정보		
패키지 및 영역에 대한 자세한 설명	System Administration Guide: Solaris Containers-Resource Management and Solaris Zones 의 24 장, “About Packages and Patches on a Solaris System With Zones Installed (Overview)”		

설명	자세한 정보
희소 및 전체 루트 영역의 개요	System Administration Guide: Solaris Containers-Resource Management and Solaris Zones 의 16 장, “Introduction to Solaris Zones”
패키지 특성 및 매개 변수에 대한 설명	pkginfo(4)
패키지 매개 변수 값 표시에 대한 설명	pkgparam(1)

배경 정보

다음 참조에는 패키지 요구 사항과 특정 명령 구문에 대한 배경 정보가 있습니다.

패키지 요구 사항 및 용어 정의에 대한 구체적인 내용	Application Packaging Developer’s Guide 의 6 장, “Advanced Techniques for Creating Packages”
패키지와 설치 관리 파일의 추가 및 제거에 대한 기본 설명	System Administration Guide: Basic Administration 의 19 장, “Managing Software (Overview)”
이 부록에서 참조하는 특정 명령에 대한 자세한 내용은 다음 매뉴얼 페이지를 참조하십시오.	dircmp(1), fssnap(1M), ps(1) 또는 truss(1) pkgadd(1M), pkgchk(1M) 또는 pkgrm(1M)
Solaris Live Upgrade 개요	2 장, “Solaris Live Upgrade(개요)”
사용자 정의 JumpStart 개요	Solaris 10 5/09 설치 설명서: 사용자 정의 JumpStart 및 고급 설치 의 2 장, “사용자 정의 JumpStart(개요)”
Solaris 영역 개요	System Administration Guide: Solaris Containers-Resource Management and Solaris Zones 의 16 장, “Introduction to Solaris Zones”

업그레이드할 때 패치 분석기 사용(작업)

이 장에서는 Solaris OS를 업그레이드하기 전에 패치 분석기를 사용하여 패치를 검사하는 지침을 제공합니다. 초기 Solaris 10 3/05 릴리스 이후 출시된 다음 릴리스 중 하나로 업그레이드할 경우 패치 분석기가 사용자 시스템에 대한 분석을 수행합니다.

- Solaris 10 1/06 릴리스
- Solaris 10 6/06 릴리스
- Solaris 10 11/06
- Solaris 10 8/07
- Solaris 10 5/08
- Solaris 10 10/08

Solaris 업데이트 릴리스로 업그레이드

10 3/05 릴리스 OS를 이미 실행하는 중이고 개별 패치를 설치한 후 후속 Solaris 10 릴리스로 업그레이드하면 다음과 같은 문제가 발생할 수 있습니다.

- 위에 명시한 릴리스 중 하나의 일부로 제공된 모든 패치가 시스템에 다시 적용됩니다. 이 패치들은 제거할 수 없습니다.
- 이전에 시스템에 설치된 패치 중에 위에 명시된 릴리스 중 하나에 포함되어 있지 않은 패치는 모두 제거됩니다.

패치 분석기는 사용자의 시스템을 분석하여 패치가 있을 경우, 위 릴리스로의 업그레이드에 의해 옮겨질 패치를 파악합니다. 패치 분석기는 다음 형식으로 사용할 수 있습니다.

- Solaris 설치 프로그램을 사용하여 업그레이드하면 패치 분석기 대화 상자가 나타납니다. 분석을 수행하려면 예를 선택합니다.
- 텍스트 설치 프로그램을 사용하여 업그레이드하려면 Analyze Patches(패치 분석) 대화 상자에서 Analyze(분석)를 선택하여 분석을 수행합니다.

- 사용자 정의 JumpStart 설치나 Solaris Live Upgrade를 사용하려면 `analyze_patches` 스크립트를 실행해 분석을 수행합니다. 자세한 내용은 246 페이지 “[analyze_patches 스크립트 실행](#)”을 참조하십시오.

분석을 수행한 후 패치 분석 결과에 대한 자세한 내용은 247 페이지 “[패치 분석기 출력 검토](#)”를 참조하십시오.

▼ analyze_patches 스크립트 실행

주 - `analyze_patches` 스크립트를 실행하려면 설치되어 있는 시스템과 Solaris Operating System DVD, Solaris Software CD나 네트워크 설치 이미지를 NFS 또는 로컬로 마운트된 매체를 통해 스크립트에서 액세스할 수 있어야 합니다.

1 Misc 디렉토리로 바꿉니다.

이 예에서 이미지는 로컬로 마운트된 매체에 있습니다.

```
# cd /cdrom/cdrom0/Solaris_10/Misc
```

2 analyze_patches 스크립트를 실행합니다.

```
# ./analyze_patches -R rootdir -N netdir -D databasedir
```

- R *rootdir* *rootdir*은 설치된 시스템의 루트입니다. 기본값은 /입니다.
- N *netdir* *netdir*은 설치할 OS 이미지의 루트에 대한 경로입니다. 기본값은 /cdrom/cdrom0입니다. *netdir*은 Solaris_10 디렉토리를 포함하는 디렉토리에 대한 경로입니다. NFS 마운트 지점에서 `patch_analyzer`를 실행하려면 이 옵션을 사용해야 합니다.
- D *databasedir* 스크립트를 OS 이미지의 Misc/ 디렉토리 이외의 다른 디렉토리에서 호출하면 프로그램이 패치 분석을 위해 사용하는 데이터베이스를 찾을 수 없습니다. -D 옵션을 사용하여 데이터베이스에 경로를 제공합니다. OS 이미지의 Solaris_10/Misc/database 에 위치해 있는 이 데이터베이스가 없으면 스크립트가 제대로 작동하지 않습니다.

▼ 패치 분석기 출력 검토

출력을 검토하려면 분석을 수행한 뒤 다음 절차를 사용합니다.

1 패치 분석기의 출력을 검토합니다.

패치 분석기는 다른 패치에 의해 제거, 우선 순위 감소, 축적 또는 폐기될 패치 목록을 제공합니다. 패치 축적은 패치 업그레이드와 유사합니다. 축적된 패치는 제거되고 새 패치에 의해 수정 사항이 적용됩니다. 다음과 같은 메시지가 표시됩니다.

Patch 105644-03 will be removed.

Patch 105925 will be downgraded from -02 to -01.

Patch 105776-01 will be accumulated/obsoleted by patch 105181-05.

패치 분석기 프로그램이 목록을 제공하지 않으면 시스템에 전에 설치된 패치들에 대해 아무런 조치도 취해지지 않습니다.

2 패치 교체 및 삭제 여부를 결정합니다.

- 교체 및 삭제하려면 시스템을 업그레이드합니다.
- 교체 및 삭제하지 않으려면 시스템을 업그레이드하지 않습니다.

용어집

3DES	[[Triple DES] 3중 데이터 암호화 표준). 168비트의 키 길이를 제공하는 대칭 키 암호화 방법입니다.
AES	(고급 암호화 표준) 대칭 128비트 블록 데이터 암호화 기술. 미국 정부는 2000년 10월 알고리즘의 Rijndael 변형을 암호화 표준으로 채택했습니다. AES가 정부 표준으로 DES 암호화를 대체합니다.
bootlog-cgi 프로그램	WAN 부트 설치 중 웹 서버가 원격 클라이언트 부팅 및 설치 콘솔 메시지를 수집하고 저장하도록 하는 CGI 프로그램입니다.
certstore 파일	특정 클라이언트 시스템의 디지털 인증서를 포함하는 파일입니다. SSL 협상 중에 클라이언트는 인증서 파일을 서버에 제공하라는 요청을 받을 수 있습니다. 서버에서는 이 파일을 사용하여 클라이언트 ID를 확인합니다.
CGI	Common Gateway Interface의 약자입니다. 외부 프로그램이 HTTP 서버와 통신하기 위한 인터페이스입니다. CGI를 사용하기 위해 작성한 프로그램을 CGI 프로그램 또는 CGI 스크립트라고 합니다. CGI 프로그램은 서버에서 정상적으로 처리 또는 구문 분석하지 않는 출력을 구문 분석하거나 양식을 처리합니다.
DES	Data Encryption Standard의 약자입니다. 1975년에 개발되고 1981년에 ANSI에 의해 ANSI X.3.92로 표준화된 대칭 키 암호화 방법입니다. DES에서는 56비트 키를 사용합니다.
DHCP	(동적 호스트 구성 프로토콜) 응용 프로그램 계층의 프로토콜입니다. TCP/IP 네트워크에 있는 개별 컴퓨터 또는 클라이언트가 전용의 중앙에서 관리되는 DHCP 서버에서 IP 주소와 다른 네트워크 구성 정보를 추출할 수 있습니다. DHCP는 대형 IP 네트워크를 유지 및 관리하는 총 비용을 절감시켜 줍니다.
암호화	정보를 알아볼 수 없도록 하여 권한이 없는 사용자가 정보를 사용하지 못하게 하는 과정입니다. 암호화는 정보를 해독하는 데 사용되는 키라고 하는 코드를 기반으로 합니다. 또한 암호 해독을 참조하십시오.
/etc/netboot 디렉토리	WAN 부트 설치에 필요한 클라이언트 구성 정보와 보안 데이터가 포함된 WAN 부트 서버의 디렉토리입니다.
/etc 디렉토리	핵심 시스템 구성 파일과 유지 보수 명령을 포함하는 디렉토리입니다.
/export 파일 시스템	네트워크상의 다른 시스템과 공유되는 OS 서버의 파일 시스템입니다. 예를 들어, /export 파일 시스템은 디스크가 없는 클라이언트를 위한 루트(/) 파일 시스템 및 스왑 공간과 네트워크 사용자를 위한 홈 디렉토리를 포함할 수 있습니다. 디스크 없는 클라이언트는 부트 및 실행을 위해 OS 서버에 있는 /export 파일 시스템에 의존합니다.

fdisk 분할 영역	x86 기반 시스템의 특정 운영 체제 전용인 디스크 드라이브의 논리적 분할 영역입니다. Solaris 소프트웨어를 설치하려면 x86 기반 시스템에 하나 이상의 Solaris fdisk 분할 영역을 설정해야 합니다. x86 기반 시스템은 디스크에서 최고 4개의 fdisk 분할 영역을 허용합니다. 개별 운영 체제를 보유하기 위해 이러한 분할 영역을 사용할 수 있습니다. 각 운영 체제는 고유한 fdisk 분할 영역에 위치해야 합니다. 시스템은 디스크당 하나의 Solaris fdisk 분할 영역만 가질 수 있습니다.
파일 시스템	SunOS™ 운영 체제에서 사용자가 액세스할 수 있는 파일 및 디렉토리의 트리 구조 네트워크입니다.
format	데이터를 구조화하거나 디스크를 데이터 수신용 섹터로 나누는 작업입니다.
GRUB	x86에만 해당: GNU GRUB(GRand Unified BootLoader)는 간단한 메뉴 인터페이스가 있는 오픈 소스 부트로더입니다. 시스템에 설치된 운영 체제 목록이 메뉴에 표시됩니다. GRUB를 사용하면 Solaris OS, Linux 또는 Microsoft Windows와 같은 다양한 운영 체제를 쉽게 부트할 수 있습니다.
GRUB 기본 메뉴	x86에만 해당: 시스템에 설치된 운영 체제 목록을 표시하는 부트 메뉴입니다. 이 메뉴에서는 BIOS 또는 fdisk 분할 영역 설정을 수정하지 않고도 운영 체제를 쉽게 부트할 수 있습니다.
GRUB 편집 메뉴	x86에만 해당: GRUB 기본 메뉴의 하위 메뉴인 부트 메뉴입니다. 이 메뉴에는 GRUB 명령이 표시됩니다. 이러한 명령을 편집하여 부트 동작을 변경할 수 있습니다.
HMAC	메시지 인증을 위해 입력한 해시 방법입니다. HMAC는 비밀 공유 키와 조합하여 MD5 또는 SHA-1과 같은 반복 암호화 해시 기능과 함께 사용합니다. 기본 해시 기능의 등록 정보에 따라 HMAC의 암호화 강도가 달라집니다.
HTTP	HyperText Transfer Protocol(n.)의 약자입니다. 원격 호스트에서 하이퍼텍스트 객체를 불러오는 인터넷 프로토콜입니다. 이 프로토콜은 TCP/IP를 기반으로 합니다.
HTTPS	HTTP의 보안 버전으로 SSL(Secure Sockets Layer)을 사용하여 구현합니다.
IPv6	IPv6은 현재 버전인 IPv4(버전 4)에서 혁신적으로 향상이 이루어지도록 설계된 새로운 버전(버전 6)의 인터넷 프로토콜(IP)입니다. 정의된 전환 메커니즘을 사용하여 IPv6을 배치하면 현재 작동이 중단되지 않습니다. 또한 IPv6은 새 인터넷 기능을 위한 플랫폼을 제공합니다.
JumpStart 디렉토리	사용자 정의 JumpStart 설치용 프로파일 디스켓을 사용할 때 JumpStart 디렉토리는 모든 핵심 사용자 정의 JumpStart 파일을 포함하는 디스켓상의 루트 디렉토리입니다. 사용자 정의 JumpStart 설치용 프로파일 서버를 사용할 때 JumpStart 디렉토리는 모든 핵심 사용자 정의 JumpStart 파일을 포함하는 서버상의 디렉토리입니다.
JumpStart 설치	출하시 설치된 JumpStart 소프트웨어를 사용하여 Solaris 소프트웨어가 자동으로 설치되는 형태의 설치입니다.
keystore 파일	클라이언트와 서버에서 공유하는 키가 포함된 파일입니다. WAN 부트 설치 중에 클라이언트 시스템은 키를 사용하여 서버에서 전송된 데이터와 파일을 해독하거나 무결성을 검증합니다.
LAN	Local Area Network의 약자입니다. 하드웨어 및 소프트웨어를 연결하여 통신할 수 있는 근거리 컴퓨터 시스템 그룹입니다.

LDAP	(Lightweight 디렉토리 액세스 프로토콜) LDAP 이름 지정 서비스 클라이언트와 서버가 서로 통신하는데 사용하는 표준의 확장 가능 디렉토리 액세스 프로토콜입니다.
MD5	(메시지 다이제스트 5) 전자 서명을 포함하여 메시지 인증용으로 사용되는 반복적인 암호화 해시 기능입니다. 이 기능은 1991년 Rivest가 개발했습니다.
menu.lst 파일	x86에만 해당: 시스템에 설치된 모든 운영 체제를 나열하는 파일입니다. 이 파일의 내용에 따라 GRUB 메뉴에 표시되는 운영 체제 목록이 결정됩니다. BIOS 또는 fdisk 분할 영역 설정을 수정하지 않고 GRUB 메뉴에서 운영 체제를 쉽게 부트할 수 있습니다.
NIS	SunOS 4.0(최소) 네트워크 정보 서비스입니다. NIS는 네트워크상의 시스템과 사용자에게 대한 핵심 정보를 포함하는 분산 네트워크 데이터베이스입니다. NIS 데이터베이스는 마스터 서버와 모든 슬레이브 서버에 저장됩니다.
NIS+	SunOS 5.0(최소) 네트워크 정보 서비스입니다. NIS+는 NIS, SunOS 4.0(최소) 네트워크 정보 서비스를 대체합니다.
/opt 파일 시스템	타사 소프트웨어 및 번들화되지 않은 소프트웨어에 대한 마운트 지점을 포함하는 파일 시스템입니다.
OS 서버	네트워크상의 시스템에 서비스를 제공하는 시스템입니다. OS 서버가 디스크가 없는 클라이언트에 서비스를 제공하려면 디스크가 없는 각 클라이언트의 루트(/) 파일 시스템과 스왑 공간(/export/root, /export/swap)을 위한 디스크 공간을 별도로 확보해야 합니다.
RAID-0 볼륨	스트라이프 또는 연결일 수 있는 볼륨 클래스입니다. 이 구성 요소는 하위미러라고도 합니다. 스트라이프 또는 연결은 미러의 기본 빌딩 블록입니다.
RAID-1 볼륨	여러 복사본을 유지하여 데이터를 복제하는 볼륨 클래스입니다. RAID-1 볼륨은 하위 미러 라고 하는 하나 이상의 RAID-0 볼륨으로 구성됩니다. RAID-1 볼륨을 미러 라고도 합니다.
RAID-Z 저장소 풀	ZFS 저장소 풀로 사용 가능한 여러 디스크에 데이터 및 패리티를 저장하는 가상 장치입니다. RAID-Z는 RAID-5와 유사합니다.
rules.ok 파일	rules 파일의 생성된 버전입니다. 사용자 정의 JumpStart 설치 소프트웨어가 시스템을 프로필에 일치시키기 위해서는 rules.ok 파일이 필요합니다. rules.ok 파일을 만들기 위해서는 반드시 check 스크립트를 사용해야 합니다.
rules 파일	자동으로 설치하려는 각 시스템 그룹(또는 단일 시스템)에 대한 규칙을 포함하는 텍스트 파일입니다. 각 규칙은 하나 이상의 시스템 속성에 기반한 시스템 그룹을 구별합니다. rules 파일은 각 그룹을 프로필에 연결하며, 이 프로필은 Solaris 소프트웨어가 그룹의 각 시스템에 설치되는 방식을 정의하는 텍스트 파일입니다. rules 파일은 사용자 정의 JumpStart 설치에 사용됩니다. 프로필도 참조하십시오.
SHA1	(보안 해시 알고리즘) 메시지 다이제스트를 생성하기 위해 2 ⁶⁴ 보다 작은 입력 길이에서 작동하는 알고리즘입니다.
Solaris DVD 또는 CD 이미지	시스템에 설치된 Solaris 소프트웨어로 Solaris DVD 또는 CD에서 액세스하거나 Solaris DVD 또는 CD 이미지에서 복사한 설치 서버의 하드 디스크에서 액세스할 수 있습니다.

Solaris Flash	사용자가 마스터 시스템 이라고 하는 시스템에서 파일의 아카이브를 만들 수 있도록 하는 Solaris 설치 프로그램입니다. 그 다음 아카이브를 다른 시스템 설치에 사용하여 다른 시스템이 구성 면에서 마스터 시스템과 동일하도록 합니다. 아카이브 도 참조하십시오.
Solaris Live Upgrade	활성화된 부트 환경이 실행 중인 상태에서 복제 부트 환경을 업그레이드할 수 있어 생산 환경의 중단 시간을 없앨 수 있는 업그레이드 방법입니다.
Solaris 설치 프로그램	그래픽 사용자 인터페이스(GUI) 또는 명령줄 인터페이스(CLI) 설치 프로그램으로 마법사 패널을 사용하여 Solaris 소프트웨어 및 타사 소프트웨어 설치를 단계별로 안내합니다.
Solaris 영역	운영 체제 서비스를 가상화하고 실행 중인 응용 프로그램을 위한 격리되고 안전한 환경을 제공하는 데 사용되는 소프트웨어 영역 분할 기술입니다. 비전역 영역을 만드는 경우 다른 모든 영역으로부터 프로세스를 격리하는 응용 프로그램 실행 환경을 만들게 됩니다. 이렇게 격리하면 한 영역에서 실행되는 프로세스가 다른 영역에서 실행되는 프로세스를 모니터링하거나 영향을 미치지 않도록 방지합니다. 전역 영역 및 비전역 영역 을 참조하십시오.
SSL (Secure Sockets Layer)	보안 버전의 HTTP인 HTTPS를 구현하기 위해 사용한 두 당사자(클라이언트와 서버)간의 보안 연결을 설정하는 소프트웨어 라이브러리입니다.
sysidcfg 파일	시스템을 미리 구성하는 특수 시스템 구성 키워드 집합을 지정하는 파일입니다.
truststore 파일	하나 이상의 디지털 인증서가 포함된 파일입니다. WAN 부트 설치 중에 클라이언트 시스템은 truststore 파일의 데이터를 확인하여 설치를 시도하는 서버의 아이디를 확인합니다.
URL	Uniform Resource Locator의 약자입니다. 문서를 요청하기 위해 서버와 클라이언트에서 사용한 주소 지정 시스템입니다. URL을 위치라고도 합니다. URL의 형식은 <i>protocol://machine:port/document</i> 입니다. 예제 URL은 http://www.example.com/index.html 입니다.
/usr 파일 시스템	많은 표준 UNIX 프로그램을 포함하는 독립 실행형 시스템 또는 서버의 파일 시스템입니다. 로컬 복사본을 유지하지 않고 큰 /usr 파일 시스템을 공유하면 시스템에서 Solaris 소프트웨어를 설치하고 실행하는 데 요구되는 전체 디스크 공간을 최소화할 수 있습니다.
/var 파일 시스템	시스템에서 계속 변경 또는 증가될 가능성이 있는 시스템 파일을 포함하는 파일 시스템 또는 디렉토리(독립 실행형 시스템의 경우)입니다. 이러한 파일로는 시스템 로그, vi 파일, 메일 파일, UUCP 파일 등이 있습니다.
WAN	(Wide Area Network) 전화, 광섬유 또는 위성 연결을 통하여 서로 다른 지리적 위치에 있는 LAN(Local Area Network) 또는 시스템을 연결하는 네트워크입니다.
WAN 부트 미니루트	WAN 부트 설치를 수행할 수 있도록 수정된 미니루트입니다. WAN 부트 미니루트에는 Solaris 미니루트에 있는 소프트웨어의 일부가 들어 있습니다. 또한 미니루트 를 참조하십시오.
WAN 부트 서버	WAN 부트 설치 중에 사용한 구성 및 보안 파일을 제공하는 웹 서버입니다.
WAN 부트 설치	HTTP 또는 HTTPS를 사용하여 WAN(wide area network)에서 소프트웨어를 부트 및 설치할 수 있게 해 주는 설치 유형입니다. WAN 부트 설치 방법을 사용하면 공용 네트워크에서 암호화된 Solaris Flash 아카이브를 전송하고 원격 클라이언트에서 사용자 정의 JumpStart 설치를 수행할 수 있습니다.

wanboot-cgi 프로그램	WAN 부트 설치에서 사용한 데이터와 파일을 검색 및 전송하는 CGI 프로그램입니다.
wanboot.conf 파일	WAN 부트 설치를 수행하는 데 필요한 구성 정보와 보안 설정을 지정하는 텍스트 파일입니다.
wanboot 프로그램	WAN 부트 설치를 수행하기 위하여 WAN 부트 미니루트, 클라이언트 구성 파일 및 설치 파일을 로드하는 제2수준 부트 프로그램입니다. WAN 부트 설치 경우 wanboot 바이너리가 ufsboot 또는 inetboot 제2수준 부트 프로그램과 비슷한 작업을 수행합니다.
ZFS	물리적 저장소를 관리하기 위해 저장소 풀을 사용하는 파일 시스템입니다.
가상 장치	ZFS 풀의 논리적 장치로 물리적 장치, 파일 또는 장치 모음으로 사용될 수 있습니다.
개발자 Solaris 소프트웨어 그룹	최종 사용자 Solaris 소프트웨어 그룹과 라이브러리를 포함하고 파일, 매뉴얼 페이지 및 소프트웨어 개발용 프로그래밍 도구를 포함하는 소프트웨어 그룹입니다.
개인 키	공용 키 암호화에 사용한 암호 해독 키입니다.
검사 키워드	사용자 정의 JumpStart 방법을 사용하여 설치할 때 시스템에 대한 속성 정보를 추출하는 구문 요소입니다. 검사 키워드를 사용하는 경우 일치 조건을 설정하고 규칙에 필요한 프로필을 실행할 필요가 없습니다. 규칙도 참조하십시오.
공용 키	공용 키 암호화에 사용한 암호 키입니다.
공용 키 암호화 도구	모든 사용자에게 알려진 공용 키와 메시지 수신자에게만 알려진 개인 키를 사용하는 암호 체계입니다.
공유 가능한 파일 시스템	/export/home 및 /swap과 같은 사용자 정의 파일인 파일 시스템입니다. 이러한 파일 시스템은 Solaris Live Upgrade를 사용할 때 활성 및 비활성 부트 환경 사이에서 공유됩니다. 공유 가능 파일 시스템은 활성 및 비활성 부트 환경 모두에서 vfstab 파일에 동일한 마운트 지점을 포함합니다. 활성 부트 환경에서 공유된 파일을 업데이트하면 비활성 부트 환경에서도 데이터가 업데이트됩니다. 공유 가능 파일 시스템은 기본적으로 공유되지만 대상 슬라이스를 지정할 수 있고 그런 다음 파일 시스템이 공유됩니다.
규칙	프로필에 하나 이상의 시스템 속성을 할당하는 일련의 값입니다. 규칙은 사용자 정의 JumpStart 설치에 사용됩니다.
기능 키	특정 작업에 매핑된, F1, F2, F3 등의 레이블이 붙은 10개 이상의 키보드 키 중 하나입니다.
네트워크 설치	네트워크를 통해 CD-ROM 또는 DVD-ROM 드라이브를 가진 시스템에서 CD-ROM이나 DVD-ROM 드라이브가 없는 시스템으로 소프트웨어를 설치하는 방법입니다. 네트워크 설치를 수행하려면 이름 서버 및 설치 서버가 필요합니다.
네트워크로 연결된 시스템	하드웨어 및 소프트웨어를 통하여 연결되어 정보를 교환 및 공유하도록 하는 시스템(호스트라고 함)의 그룹입니다. LAN(local area network)이라고 합니다. 시스템이 네트워크에 연결되면 일반적으로 하나 이상의 서버가 필요합니다.
네트워크에 연결되지 않은 시스템	네트워크에 연결되지 않았거나 다른 시스템에 의존하지 않는 시스템입니다.

논리적 장치	시스템에는 하나의 장치로 보이는 한 개 이상의 디스크에 있는 물리적 슬라이스의 그룹입니다. 논리적 장치는 Solaris 볼륨 관리자에서 볼륨이라고 합니다. 볼륨은 응용 프로그램 또는 파일 시스템의 관점에서 보면 물리적 디스크와 기능적으로 동일합니다.
데이터 세트	복제, 파일 시스템, 스냅샷 또는 볼륨과 같은 ZFS 엔티티의 일반적인 이름입니다.
도메인	인터넷 이름 지정 계층의 일부입니다. 도메인은 관리 파일을 공유하는 로컬 네트워크상의 시스템 그룹을 표현합니다.
도메인 이름	관리 파일을 공유하는 로컬 네트워크상의 시스템 그룹에 할당된 이름입니다. 네트워크 정보 서비스(NIS) 데이터베이스가 제대로 작동하려면 도메인 이름이 필요합니다. 도메인 이름은 마침표로 구분되는 구성 요소 이름의 시퀀스로 구성됩니다(예: tundra.mpk.ca.us). 도메인 이름을 왼쪽에서 오른쪽으로 읽음에 따라 구성 요소 이름은 관리 기관의 보다 일반적인(일반적으로 원격) 영역을 식별합니다.
독립 실행형	다른 모든 시스템의 지원을 필요로 하지 않는 컴퓨터입니다.
디스크 (disc)	자기 디스크와 구분되는 광 디스크로서 CD(컴팩트 디스크) 시장에서 사용되는 일반적인 철자를 인식합니다. 예를 들어, CD-ROM 또는 DVD-ROM은 광 디스크입니다.
디스크 (disk)	파일과 같이 데이터를 저장하기 위한 집중 트랙과 섹터로 구성되는 자기화된 매체의 둥근 레코드 또는 레코드 세트입니다. 디스크(disk)도 참조하십시오.
디스크 구성 파일	디스크의 구조를 표현하는 파일(예: 바이트/섹터, 플래그, 슬라이스)입니다. 디스크 구성 파일을 이용하면 단일 시스템에서 pfinstall 명령을 사용하여 여러 크기의 디스크에서 프로파일을 테스트할 수 있습니다.
디스크 없는 클라이언트	모든 디스크 저장소에 대하여 서버에 의존하는 네트워크상의 클라이언트입니다.
디지털 인증서	통신 당사자가 이미 인증하는 타사에서 발급된 양도 및 위조 불가능한 디지털 파일입니다.
로컬	동일한 언어, 관습 또는 문화적 관례(미국용 영어는 en_US, 영국용 영어는 en_UK)를 공유하는 지리적 또는 정치적 지역 또는 커뮤니티입니다.
루트	항목 계층의 최상위 수준입니다. 루트는 다른 모든 항목이 시작되는 항목입니다. 루트 디렉토리 또는 루트(/) 파일 시스템을 참조하십시오.
루트 (/) 파일 시스템	다른 모든 파일 시스템이 시작되는 최상위 파일 시스템입니다. 루트(/) 파일 시스템은 다른 모든 파일 시스템이 마운트되는 기반이며 마운트 해제되지 않습니다. 루트(/) 파일 시스템은 시스템 시작(부트) 시 사용되는 커널, 장치 드라이버 및 프로그램과 같이 시스템 작업에 핵심적인 디렉토리 및 파일을 포함합니다.
루트 디렉토리	다른 모든 디렉토리가 시작되는 최상위 디렉토리입니다.
마스터 시스템	Solaris Flash 아카이브를 설치하는 데 사용되는 시스템입니다. 시스템 구성은 아카이브에 저장됩니다.
마운트	마운트 요청을 하는 시스템 또는 네트워크상의 원격 디스크에 첨부한 디스크에서 디렉토리에 액세스하는 프로세스입니다. 파일 시스템을 마운트하려면 로컬 시스템의 마운트 지점과 마운트될 파일 시스템의 이름(예: /usr)이 필요합니다.
마운트 지점	원격 시스템에 존재하는 파일 시스템을 마운트하는 워크스테이션 디렉토리입니다.

마운트 해제	네트워크상의 컴퓨터 또는 원격 디스크에 연결된 디스크의 디렉토리에 대한 액세스 권한을 제거하는 프로세스입니다.
매니페스트 섹션	복제 시스템의 유효성을 확인하기 위해 사용되는 Solaris Flash 아카이브의 섹션입니다. 매니페스트 섹션은 복제 시스템에서 보존, 추가 또는 삭제된 시스템상의 파일을 나열합니다. 이 섹션은 정보에 불과합니다. 이 섹션은 내부 형식을 사용하여 파일을 나열하며 스크립트에는 사용할 수 없습니다.
매체 서버	설치 서버를 참조하십시오.
메타 장치	볼륨을 참조하십시오.
명령줄	명령으로 시작하는 문자열로서 흔히 옵션, 파일 이름 및 기타 표현식을 포함하는 인자가 뒤에 오며 행 끝 문자로 끝납니다.
문서 루트 디렉토리	서버에 액세스하는 사용자에게 제시할 파일, 이미지 및 데이터 등이 들어있는 서버 컴퓨터에 있는 계층의 루트입니다.
미니루트	Solaris 설치 매체에 포함된 최소 부트 가능 루트(/) 파일 시스템입니다. 미니루트는 시스템을 설치 및 업그레이드하는 데 필요한 Solaris 소프트웨어로 구성됩니다. x86 기반 시스템에서 미니루트는 비상 안전 부트 아카이브로 사용될 시스템으로 복사됩니다. 비상 안전 부트 아카이브 를 참조하십시오.
미러	RAID-1 볼륨을 참조하십시오.
복제 시스템	Solaris Flash 아카이브를 사용하여 설치한 시스템입니다. 복제 시스템은 마스터 시스템과 동일한 설치 구성을 갖습니다.
볼륨	시스템에 하나의 논리적 장치로 보이는 물리적 슬라이스 또는 다른 볼륨의 그룹입니다. 볼륨은 응용 프로그램 또는 파일 시스템의 관점에서 보면 물리적 디스크와 기능적으로 동일합니다. 일부 명령줄 유틸리티에서는 볼륨을 메타 장치라고도 합니다. 볼륨은 표준 UNIX 용어로 의사 장치 또는 가상 장치 라고도 합니다.
볼륨 관리자	DVD-ROM, CD-ROM 및 디스켓의 데이터에 대한 액세스 권한을 관리하고 얻는 체계를 제공하는 프로그램입니다.
부트	시스템 소프트웨어를 메모리에 로드하고 시스템 소프트웨어를 시작하는 작동입니다.
부트 로더	x86에만 해당: 부트 로더는 시스템을 켜 후에 실행되는 첫 번째 소프트웨어 프로그램입니다. 이 프로그램이 부팅 프로세스를 시작합니다.
부트 서버	동일한 네트워크 서브넷의 클라이언트 시스템에 해당 클라이언트 시스템이 시작해야 하는 프로그램 및 정보를 제공하는 서버 시스템입니다. 부트 서버는 설치 서버가 Solaris 소프트웨어가 설치될 시스템과 다른 서브넷에 있을 경우 네트워크를 통해 설치해야 합니다.

부트 아카이브

x86에만 해당: 부트 아카이브는 Solaris OS를 부트하는 데 사용되는 핵심 파일 모음입니다. 이러한 파일은 루트(/) 파일 시스템이 마운트되기 전에 시스템 시작 중에 필요합니다. 다음 두 개의 부트 아카이브가 시스템에서 유지 관리됩니다.

- 시스템에서 Solaris OS를 부트하는 데 사용되는 부트 아카이브. 이 부트 아카이브를 주 부트 아카이브라고도 합니다.
- 주 부트 아카이브가 손상된 경우 복구에 사용되는 부트 아카이브. 이 부트 아카이브는 루트(/) 파일 시스템을 마운트하지 않고 시스템을 시작합니다. GRUB 메뉴에서는 이 부트 아카이브를 비상 안전이라고 합니다. 이 아카이브의 기본 목적은 일반적으로 시스템 부트 시 사용되는 주 부트 아카이브를 재생성하는 것입니다.

부트 환경

Solaris OS의 운영에 중요한 필수 파일 시스템(디스크 슬라이스 및 마운트 지점)의 모음입니다. 이 디스크 슬라이스들은 동일한 디스크에 있을 수도 있고 여러 디스크에 분산되어 있을 수 있습니다.

활성 부트 환경은 현재 부트된 환경입니다. 정확히 하나의 활성 부트 환경만 부트될 수 있습니다. 비활성 부트 환경은 현재 부트되지 않았지만 다음 재부트 시 작동 대기 상태일 수 있습니다.

비상 안전 부트 아카이브

x86에만 해당: 주 부트 아카이브가 손상된 경우 복구에 사용되는 부트 아카이브입니다. 이 부트 아카이브는 루트(/) 파일 시스템을 마운트하지 않고 시스템을 시작합니다. GRUB 메뉴에서는 이 부트 아카이브를 비상 안전이라고 합니다. 이 아카이브의 기본 목적은 일반적으로 시스템 부트 시 사용되는 주 부트 아카이브를 재생성하는 것입니다. **부트 아카이브**를 참조하십시오.

비전역 영역

Solaris 운영 체제의 단일 인스턴스 내에서 만들어진 가상 운영 체제 환경입니다. 하나 이상의 응용 프로그램이 시스템의 나머지와 상호 작용하지 않고 비전역 영역에서 실행될 수 있습니다. 비전역 영역 또한 영역이라고 합니다. **Solaris 영역** 및 **전역 영역**도 참조하십시오.

사용자 정의 JumpStart

Solaris 소프트웨어가 사용자 정의 프로필을 기반으로 하는 시스템에 설치되는 형태의 설치입니다. 사용자 유형 및 시스템 유형에 따라 사용자 정의된 프로필을 만들 수 있습니다. 사용자 정의 JumpStart 설치하는 사용자가 생성한 JumpStart 설치입니다.

사용자 정의 검사 파일

동일한 JumpStart 디렉토리에 rules 파일로 위치해야 하는 파일로서 검사 및 비교라는 두 가지 함수 유형을 포함하는 Bourne 셸 스크립트입니다. 검사 및 비교. 검사 함수는 원하는 정보를 모으거나 실제 작업을 수행하고 사용자가 정의한 해당 **SI_** 환경 변수를 설정합니다. 검사 함수는 검사 키워드가 됩니다. 비교 함수는 해당하는 검사 함수를 호출하고, 검사 함수의 출력을 비교하고, 키워드가 일치하면 0을 반환하고 키워드가 일치하지 않으면 1을 반환합니다. 비교 함수는 규칙 키워드가 됩니다. 또한 **rules** 파일을 참조하십시오.

상태 데이터베이스

Solaris Volume Manager 구성의 상태에 대한 정보를 저장하는 데이터베이스입니다. 상태 데이터베이스는 복제된 여러 데이터베이스 복사본의 모음입니다. 각 복사본을 **상태 데이터베이스 복제본**이라고 합니다. 이 상태 데이터베이스는 모든 알려진 상태 데이터베이스 복제본의 위치 및 상태를 추적합니다.

상태 데이터베이스 복제본

상태 데이터베이스의 복사본입니다. 복제본은 데이터베이스에 있는 데이터가 유효한지 확인합니다.

서버

자원을 관리하고 클라이언트에 서비스를 제공하는 네트워크 장치입니다.

서브넷	경로 지정을 단순화하기 위해 단일 논리적 네트워크를 더 작은 물리적 네트워크로 나누는 작동 체계입니다.
서브넷 마스크	서브넷 주소 지정을 위해 인터넷 주소에서 비트를 선택하는 데 사용되는 비트 마스크입니다. 마스크는 32비트 길이이고, 인터넷 주소의 네트워크 부분과 로컬 부분의 1 또는 그 이상의 비트를 선택합니다.
설치 서버	네트워크상의 다른 시스템에서 Solaris를 설치할 수 있도록 Solaris DVD 또는 CD 이미지를 제공하는 서버(매체 서버 라고도 함)입니다. Solaris DVD 또는 CD 이미지를 서버의 하드 디스크에 복사하여 설치 서버를 만들 수 있습니다.
소프트웨어 그룹	Solaris 소프트웨어의 논리적 그룹(클러스터 및 패키지)입니다. Solaris를 설치하는 동안 코어, 최종 사용자 Solaris 소프트웨어, 개발자 Solaris 소프트웨어 또는 전체 Solaris 소프트웨어 및 SPARC 시스템 전용, 전체 Solaris 소프트웨어 그룹 및 OEM 지원.
수퍼유저	시스템에서 모든 관리 작업을 수행할 수 있는 권한을 가진 특수 사용자입니다. 수퍼유저는 모든 파일을 읽거나 쓰고, 모든 프로그램을 실행하며, 모든 프로세스에 종료 신호를 보낼 수 있습니다.
스냅샷	지정된 시점의 ZFS 파일 시스템 또는 볼륨의 읽기 전용 이미지입니다.
스왑 공간	메모리 영역의 내용을 메모리로 다시 로드할 수 있을 때까지 해당 내용을 임시로 보유하는 슬라이스 또는 파일입니다. /swap 또는 swap 볼륨이라고도 합니다.
슬라이스	디스크 공간이 소프트웨어에 의해 나뉘는 단위입니다.
시스템 구성 파일	(system.conf) WAN 부트 설치에서 사용할 sysidcfg 파일 및 사용자 정의 JumpStart 파일 위치를 지정하는 텍스트 파일입니다.
시작 스크립트	rules 파일 내에 지정된 사용자 정의 Bourne 셸 스크립트로서 시스템에 Solaris 소프트웨어가 설치되기 전에 작업을 수행합니다. 사용자 정의 JumpStart 설치에 한하여 시작 스크립트를 사용할 수 있습니다.
아카이브	마스터 시스템에서 복사한 파일의 모음을 포함하는 파일입니다. 이 파일에는 이름 및 아카이브를 만든 날짜 등과 같은 아카이브에 대한 식별 정보도 포함되어 있습니다. 시스템에 아카이브를 설치하면 해당 시스템이 마스터 시스템의 정확한 구성을 갖게 됩니다. 아카이브는 차등 아카이브가 될 수 있습니다. 차등 아카이브는 원래의 마스터 이미지와 업데이트된 마스터 이미지라는 두 시스템 이미지 간의 차이만을 포함하는 Solaris Flash 아카이브입니다. 차등 아카이브에는 복제 시스템에서 보존, 수정 또는 삭제된 파일이 포함되어 있습니다. 차등 업데이트는 지정된 파일만을 변경하며, 원래의 마스터 이미지와 일치하는 소프트웨어가 포함된 시스템에만 제한됩니다.
암호 해독	암호화된 데이터를 일반 텍스트로 변환하는 과정입니다. 또한 암호화 를 참조하십시오.
업그레이드	파일을 기존 파일과 병합하고 가능한 경우 수정 사항을 보존하는 설치 방법입니다. Solaris OS를 업그레이드하면 새 버전의 Solaris OS와 시스템의 디스크에 있는 기존 파일을 병합합니다. 업그레이드하는 경우 이전 버전의 Solaris OS에서 적용된 수정 사항을 가능한 한 많이 유지하게 됩니다.

업그레이드 옵션	Solaris 설치에서 제공하는 옵션입니다. 업그레이드 절차에서는 새 버전의 Solaris와 디스크에 있는 기존 파일이 병합됩니다. 업그레이드는 또한 Solaris가 마지막 설치된 이후 가능한 많은 로컬 수정을 저장합니다.
업데이트	시스템에서 동일한 유형의 소프트웨어를 변경하는 설치 또는 설치 작업입니다. 업그레이드와 달리 업데이트는 시스템을 다운그레이드할 수 있습니다. 초기 설치와 달리 업데이트하려면 설치하고 있는 소프트웨어와 동일한 유형의 소프트웨어가 설치되어 있어야 합니다.
연결	RAID-0 볼륨입니다. 슬라이스가 연결되면 첫 번째 사용 가능한 슬라이스가 가득 찰 때까지 해당 슬라이스에 데이터가 기록됩니다. 슬라이스가 가득 차면 데이터는 연속적으로 다음 슬라이스에 기록됩니다. 연결은 미러에 포함된 경우를 제외하고 데이터 중복을 제공하지 않습니다. RAID-0 볼륨도 참조하십시오.
영역	비전역 영역 을 참조하십시오.
유틸리티	컴퓨터의 관리를 수행하는, 일반적으로 컴퓨터 구입과 함께 추가 비용 없이 제공되는 표준 프로그램입니다.
이름 서버	네트워크상의 시스템에 이름 지정 서비스를 제공하는 서버입니다.
이름 지정 서비스	시스템 간 상호 통신할 수 있도록 네트워크상의 모든 시스템에 대한 핵심 시스템 정보를 포함하는 분산 네트워크 데이터베이스입니다. 이름 지정 서비스를 사용하여 네트워크상에서 시스템 정보를 유지, 관리 및 액세스할 수 있습니다. 이름 지정 서비스가 없으면 각 시스템은 시스템 정보의 자체 복사본을 /etc 파일에 유지해야 합니다. Sun은 LDAP, NIS 및 NIS+ 이름 지정 서비스를 제공합니다.
인증기관	(CA) 전자 서명과 공용-개인 키 쌍을 만드는데 사용하는 전자 인증서를 발급하는 신뢰된 제3자 단체 또는 회사입니다. CA는 고유한 인증서를 부여받은 개인의 자격을 증명합니다.
작업	컴퓨터 시스템에 의해 완료될 사용자 정의 작업입니다.
전역 영역	Solaris 영역에서 전역 영역은 시스템의 기본 영역이며 또한 시스템 전체의 관리 제어용으로 사용되는 영역입니다. 전역 영역은 비전역 영역을 구성, 설치, 관리 또는 제거할 수 있는 유일한 영역입니다. 실제 장치, 라우팅 또는 동적 재구성(DR) 등의 시스템 인프라에 대한 관리는 전역 영역에서만 가능합니다. 전역 영역에서 적절한 권한으로 실행되는 프로세스는 다른 영역에 연결된 객체에 액세스할 수 있습니다. Solaris 영역 및 비전역 영역 을 참조하십시오.
전원 관리	시스템의 유휴 상태가 30분 이상 지속되면 시스템 상태를 자동 저장한 뒤 시스템을 끄는 소프트웨어입니다. 미국 환경 보호 기관의 Energy Star 지침의 버전 2를 준수하는 시스템에 Solaris 소프트웨어를 설치하는 경우 Power Management 소프트웨어는 기본적으로 설치됩니다. sun4u SPARC 기반 시스템은 Power Management가 기본적으로 설치되어 있는 시스템의 예입니다. 이후 재부트하면 Power Management 소프트웨어를 활성화 또는 비활성화할 것인지 묻는 메시지가 나타납니다. Energy Star 지침에 따르면 시스템 또는 모니터는 비활성화된 후 자동으로 "휴면 상태"(30와트 미만의 전력 소모)에 들어가야 합니다.
전체 Solaris 소프트웨어 그룹	전체 Solaris 릴리스가 포함된 소프트웨어 그룹입니다.

전체 Solaris 소프트웨어 그룹 및 OEM 지원	전체 Solaris 릴리스와 OEM에 대한 추가 하드웨어 지원이 포함된 소프트웨어 그룹입니다. 이 소프트웨어 그룹은 SPARC 기반 서버에 Solaris 소프트웨어를 설치할 때 권장됩니다.
종료 스크립트	rules 파일 내에 지정되며, Solaris 소프트웨어가 시스템에 설치된 후 그러나 시스템 재부트 전에 작업을 수행하는 사용자 정의 Bourne 셸 스크립트. 사용자 정의 JumpStart 설치와 함께 종료 스크립트를 사용할 수 있습니다.
주 부트 아카이브	시스템에서 Solaris OS를 부트하는 데 사용되는 부트 아카이브입니다. 이 부트 아카이브를 주 부트 아카이브라고도 합니다. 부트 아카이브 를 참조하십시오.
중요 파일 시스템	Solaris OS에서 필요한 파일 시스템입니다. Solaris Live Upgrade를 사용할 때 이 파일 시스템은 활성 및 비활성 부트 환경의 <code>vfstab</code> 파일에서 별도의 마운트 지점입니다. 파일 시스템의 예로 <code>root(/)</code> , <code>/usr</code> , <code>/var</code> , <code>/opt</code> 등이 있습니다. 이 파일 시스템은 항상 소스에서 비활성 부트 환경으로 복사됩니다.
차등 아카이브	원래의 마스터 이미지와 업데이트된 마스터 이미지라는 두 시스템 이미지 간의 차이만을 포함하는 Solaris Flash 아카이브입니다. 차등 아카이브에는 복제 시스템에서 보존, 수정 또는 삭제된 파일이 포함되어 있습니다. 차등 업데이트는 지정된 파일만 변경하며 원래의 마스터 이미지와 일관된 소프트웨어를 포함하는 시스템으로만 제한됩니다.
체크섬	그룹 검사에 사용되는 데이터 항목 그룹을 추가한 결과입니다. 데이터 항목은 숫자이거나, 체크섬 계산 동안 숫자로 처리되는 다른 문자열일 수 있습니다. 체크섬 값은 두 장치 사이의 통신이 성공적임을 나타냅니다.
초기 설치	현재 실행되는 소프트웨어를 덮어쓰거나 빈 디스크를 초기화하는 설치입니다. Solaris OS를 초기 설치하면 시스템의 디스크 또는 새 버전의 Solaris OS가 있는 디스크를 덮어쓰게 됩니다. 시스템에서 Solaris OS가 실행되지 않는 경우 반드시 초기 설치를 수행해야 합니다. 시스템에 업그레이드 버전의 Solaris OS가 실행되는 경우 초기 설치를 수행하면 디스크를 덮어쓰게 되며 OS 또는 로컬 수정 내용을 보존할 수 없습니다.
최종 사용자 Solaris 소프트웨어 그룹	공통 데스크탑 환경(CDE) 및 DeskSet 소프트웨어를 포함하여 코어 소프트웨어 그룹 및 권장되는 최종 사용자용 소프트웨어를 포함하는 소프트웨어 그룹입니다.
축소된 네트워크 지원 소프트웨어 그룹	제한적인 네트워크 서비스 지원이 있는 Solaris 시스템을 부트하고 실행하는 데 필요한 최소 코드를 포함하는 소프트웨어 그룹입니다. 네트워크 소프트웨어 그룹 감소에는 복수 사용자 텍스트 기반 콘솔 및 시스템 관리 유틸리티를 제공합니다. 이 소프트웨어 그룹을 사용하면 또한 시스템이 네트워크 인터페이스를 인식하지만 네트워크 서비스를 활성화하지는 않도록 할 수 있습니다.
커버로스	강력한 비밀번호 암호화 도구를 사용하여 클라이언트 및 서버가 비보안 네트워크 연결을 통해 서로에 대하여 자신을 식별할 수 있도록 하는 네트워크 인증 프로토콜입니다.
클라이언트	통신용 클라이언트 서버 모델에서 클라이언트는 연산 능력과 대형 메모리 용량과 같은 연산 서버의 자원에 원격으로 액세스하는 프로세스입니다.
클러스터	패키지의 논리적 모음(소프트웨어 모듈)입니다. Solaris 소프트웨어는 각각 클러스터로 구성된 소프트웨어 그룹과 패키지로 나누어집니다.
키	데이터를 암호화하거나 해독하기 위한 코드입니다. 또한 암호화 를 참조하십시오.
파생된 프로필	사용자 정의 JumpStart 설치 중에 시작 스크립트가 동적으로 만든 프로필입니다.

파일 서버	네트워크상의 시스템에 소프트웨어와 파일 저장소를 제공하는 서버입니다.
패널	창, 대화 상자 또는 애플릿의 내용으로 구성된 컨테이너입니다. 패널은 사용자 입력을 수집 및 확인할 수 있습니다. 패널은 마법사에 의해 사용될 수 있으며 지정된 작업을 완수하기 위해 순서가 정해진 시퀀스를 따릅니다.
패치 분석기	직접 실행하거나 Solaris 설치 프로그램의 일부로 실행할 수 있는 스크립트입니다. 패치 분석기는 시스템에 대한 분석을 수행하여 Solaris 업데이트로 업그레이드하여 제거할 패치(있는 경우)를 결정합니다.
패키지	모듈러 설치를 위한 단일 엔티티로 그룹화된 소프트웨어의 모음입니다. Solaris 소프트웨어는 각각 클러스터로 구성된 소프트웨어 그룹과 패키지로 나누어집니다.
폴백	이전에 실행된 환경으로의 복귀를 의미합니다. 부트 오류(또는 일부 원하지 않는 동작)에 대해 환경과 지정된 부트 환경을 활성화할 때 폴백을 사용합니다.
표준 시간대	표준 시간이 유지되는 지구 표면의 24 경도 분할 중 하나입니다.
풀	사용 가능한 ZFS 저장소의 레이아웃 및 물리적 특성을 설명하는 장치의 논리적 그룹입니다. 데이터 세트에 대한 공간이 풀에서 할당됩니다.
프로필	사용자 정의 JumpStart 방법을 사용할 때 Solaris 소프트웨어를 설치하는 방법을 정의하는 텍스트 파일입니다. 예를 들어, 프로필은 설치할 소프트웨어 그룹을 정의합니다. 모든 규칙은 규칙이 일치할 때 시스템을 설치할 방법을 정의하는 프로필을 지정합니다. 일반적으로 모든 규칙에 대하여 다른 프로필을 만듭니다. 그러나 동일한 규칙이 하나 이상의 규칙에서 사용될 수 있습니다. 또한 <i>rules</i> 파일을 참조하십시오.
프로필 디스켓	루트 디렉토리(JumpStart 디렉토리)에 모든 핵심 사용자 정의 JumpStart 파일을 포함하는 디스켓입니다.
프로필 서버	JumpStart 디렉토리에 모든 핵심 사용자 정의 JumpStart 파일을 포함하는 서버입니다.
플랫폼 그룹	특정 소프트웨어를 배포할 목적으로 공급업체가 정의한 하드웨어 플랫폼 그룹입니다. 유효한 플랫폼 그룹의 예로는 i86pc 및 sun4u가 있습니다.
플랫폼 이름	uname -i 명령의 출력입니다. 예를 들어, Ultra 60의 플랫폼 이름은 SUNW,Ultra-60입니다.
하드 링크	디스크상의 파일을 참조하는 디렉토리 항목입니다. 그러한 하나 이상의 디렉토리 항목이 동일한 물리적 파일을 참조할 수 있습니다.
하위 미러	RAID-0 볼륨을 참조하십시오.
해시	일부 입력을 가져와서 입력보다 훨씬 더 짧은 숫자를 생성하여 만든 숫자입니다. 동일한 입력에 대해 항상 동일한 출력 값이 생성됩니다. 테이블 검색 알고리즘, 오류 감지 및 무단 변경 감지에 해시 기능을 사용할 수 있습니다. 무단 변경 감지에 사용할 경우 동일한 해시 결과를 만들어내는 두 개의 입력을 찾기 어려우므로 해시 기능을 선택합니다. MD5 및 SHA-1은 단방향 해시 기능의 예입니다. 예를 들어 메시지 다이제스트는 디스크 파일 등의 가변 길이 입력을 가져와 이를 작은 값으로 줄입니다.
해싱	문자열을 원래 문자열을 나타내는 값이나 키로 변경하는 과정입니다.

핵심 소프트웨어 그룹	시스템에서 Solaris OS를 부트하고 실행하는 데 필요한 최소 소프트웨어가 포함된 소프트웨어 그룹입니다. 핵심 소프트웨어는 공통 데스크탑 환경(CDE) 데스크탑을 실행하는 데 필요한 일부 네트워크 소프트웨어 및 드라이버를 포함합니다. CDE 소프트웨어는 포함하지 않습니다.
호스트 이름	시스템이 네트워크상의 다른 시스템에 알려진 이름입니다. 이 이름은 특정 도메인 내(일반적으로 이것은 모든 단일 구성 내부를 뜻함)에서 모든 시스템 사이에 고유해야 합니다. 호스트 이름은 문자, 숫자 및 마이너스 기호(-)의 모든 조합이 될 수 있지만 마이너스 기호로 시작하거나 끝낼 수 없습니다.
화살표 키	숫자 키패드에 있는 4개의 방향 키 중 하나입니다.

색인

B

boot: cannot open /kernel/unix 메시지, 218
bootparams 파일, 업데이트, 223

C

Can't boot from file/device 메시지, 218
CHANGE DEFAULT BOOT DEVICE 메시지, 224
CLIENT MAC ADDR 오류 메시지, 223
clock gained xxx days 메시지, 218

E

/etc/bootparams 파일, JumpStart 디렉토리 액세스
활성화, 223

L

le0: No carrier - transceiver cable problem 메시지, 218
lumount 명령, 비전역 영역, 152

N

No carrier - transceiver cable problem 메시지, 218
Not a UFS filesystem 메시지, 218

R

RAID-0 볼륨, 설명, 29
RAID-1 볼륨(미러)
Solaris 볼륨 관리자 볼륨으로 마이그레이션
예, 165
만들기 및 업그레이드 예, 161
만들기 예, 80, 81, 82
설명, 27, 29
요구 사항, 48
RAID-1 볼륨(미러)에 대한 슬라이스 선택, 48
RPC 시간 초과 메시지, 222

S

Solaris Live Upgrade 명령, 169
Solaris Live Upgrade 작업 취소, 128
Solaris 볼륨 관리자
Solaris Live Upgrade에 사용되는 명령, 49
예
RAID-1 볼륨 분리 및 업그레이드, 161
RAID-1 볼륨으로 마이그레이션, 165

T

transceiver cable problem 메시지, 218

U

UFS에서 ZFS로 마이그레이션, 187
비전역 영역 포함, 207

W

WARNING: CHANGE DEFAULT BOOT DEVICE, 224
WARNING: clock gained xxx days 메시지, 218

Z

ZFS
UFS에서 ZFS로 마이그레이션, 187
개요, 173
계획, 183
마이그레이션
비전역 영역 포함, 207
부트 환경 만들기, 187
다른 소스에서, 203
동일한 풀, 194
새 풀, 198

개

개요
ZFS용, 173
그래픽, 20

계

계획, ZFS용, 183

공

공유 가능 파일 시스템, 정의, 22

내

내용 사용자 정의, 51

디

디스크 공간 요구 사항, 44

루

루트(/) 파일 시스템, 비활성 부트 환경에 대한
패키지 요구 사항, 235

만

만들기
RAID-1 볼륨(미러), 설명, 27
부트 환경, 설명, 22
부트 환경, 작업, 62, 65, 66
작업 맵, 57-58
작업
ZFS용, 187

문

문제 해결
DHCP를 사용하여 네트워크에서 부팅, 223
일반 설치 문제
DHCP를 사용하여 네트워크에서 부팅, 223
시스템 부트, 223
잘못된 서버에서 부트, 223

미

미러, 참조 RAID-1 볼륨

보

보기, 부트 환경의 구성, 136

복

복사, 파일 시스템, 127

블

블륨 관리자, 참조 Solaris 블륨 관리자

볼륨

- RAID-0, 설명, 29
- RAID-1, 설명, 29

부

- 부트 환경 비교, 129
- 부트 환경 활성화
 - 설명, 37
 - 실패, 설명, 38
 - 작업, 107
 - 파일 동기화, 설명, 52
- 부트 환경, 실패, 설명, 38
- 부트 환경의 구성 보기, 비전역 영역, 151
- 부트 환경의 이름 변경, 131

비

- 비전역 영역
 - lumount 명령, 152
 - UFS에서 ZFS로 마이그레이션, 207
 - 개요, 137
 - 그래픽, 138
 - 단계별 절차, 144
 - 별도의 파일 시스템, 143
 - 부트 환경 관리, 150
 - 업그레이드 예, 149
 - 파일 시스템 나열, 151
 - 파일 시스템 비교, 151

삭

- 삭제, 부트 환경, 130

상

- 상태 데이터베이스, 설명, 29
- 상태, 부트 환경 표시, 126

설

- 설치
 - Solaris Flash 아카이브, 102
 - 패키지, 58
 - 프로필이 있는 Solaris Flash 아카이브, 106

스

- 스왑 파일 시스템, 슬라이스 선택 지침, 50

슬

- 슬라이스, 선택 지침, 47

시

- 시간 초과 RPC 오류, 222

아

- 아카이브
 - 부트 환경 설치, 102
 - 빈 부트 환경 만들기, 75
 - 설치 예, 36

알

- 알 수 없는 클라이언트 오류 메시지, 217

업

- 업그레이드 실패
 - 복구, 115
 - 재부트 문제, 228
- 업그레이드
 - Solaris 업데이트 릴리스, 245-247
 - 비전역 영역
 - lumount 명령, 152
 - 개요, 137

업그레이드, 비전역 영역 (계속)

- 그래픽, 138
- 단계별 절차, 144
- 별도의 파일 시스템, 143
- 부트 환경 관리, 150
- 부트 환경 비교, 151
- 예, 149
- 파일 시스템 나열, 151
- 새 부트 환경
 - 부트 환경, 88
- 설명, 34
- 실패한 업그레이드 복구, 115
- 업그레이드 실패, 228
- 예, 153, 161, 165
- 작업 맵, 87-88
- 작업
 - Solaris Flash 아카이브 설치, 102
 - 지침, 88

연

- 연결, 설명, 29

예

예

- RAID-1 볼륨 만들기, 80, 81
- RAID-1 볼륨 업그레이드, 161, 165
- 내용 사용자 정의, 84, 85
- 미러 만들기, 81, 82
- 전체 프로세스
 - 전체 프로세스, 153

요

- 요구 사항, Solaris Live Upgrade 사용, 41

중

- 중요 파일 시스템, 정의, 22

키

- 키워드
 - 볼륨, 79
 - 프로필, 96, 97

테

- 테스트, 프로필, 100

토

- 토큰 링 카드, 부팅 오류, 222

파

- 파일 및 파일 시스템
 - RAID-1 볼륨(미러) 만들기, 설명, 27
 - 만들기 지침, 46
 - 부트 환경 간 파일 시스템 공유, 50
 - 설명, 22
 - 슬라이스 선택 지침, 47
 - 크기 예측, 44
- 파일 시스템 비교, 비전역 영역, 151

패

- 패치 분석기, 245-247

패치

- 추가, 45, 89
- 패치 수준 확인, 42, 61

패키지

- Solaris Live Upgrade 요구 사항, 235
- 사용자 정의 JumpStart 사용 시 요구 사항, 235
- 추가, 45, 89

표

- 표시, 부트 환경의 이름, 131

프

프로필 키워드

forced_deployment

설명 및 값, 98

local_customization

설명 및 값, 98

프로필

예, 99

차등 아카이브 예, 99

필

필수 패키지, 43

하

하위 미러, 설명, 29

