

Oracle® Solaris 10 1/13 インストールガイド: Live Upgrade とアップグレードの計画

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクル社までご連絡ください。

このソフトウェアまたは関連ドキュメントを、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供する場合は、次の通知が適用されます。

U.S. GOVERNMENT END USERS:

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

このソフトウェアもしくはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアもしくはハードウェアは、危険が伴うアプリケーション（人的傷害を発生させる可能性があるアプリケーションを含む）への用途を目的として開発されていません。このソフトウェアもしくはハードウェアを危険が伴うアプリケーションで使用する際、安全に使用するために、適切な安全装置、バックアップ、冗長性（redundancy）、その他の対策を講じることは使用者の責任となります。このソフトウェアもしくはハードウェアを危険が伴うアプリケーションで使用したこと起因して損害が発生しても、オラクル社およびその関連会社は一切の責任を負いかねます。

OracleおよびJavaはOracle Corporationおよびその関連企業の登録商標です。その他の名称は、それぞれの所有者の商標または登録商標です。

Intel、Intel Xeonは、Intel Corporationの商標または登録商標です。すべてのSPARCの商標はライセンスをもとに使用し、SPARC International, Inc.の商標または登録商標です。AMD、Opteron、AMDロゴ、AMD Opteronロゴは、Advanced Micro Devices, Inc.の商標または登録商標です。UNIXは、The Open Groupの登録商標です。

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

目次

はじめに	11
パートI Live Upgrade を使ったアップグレード	15
1 Oracle Solaris インストールの計画についての参照先	17
計画とシステム要件についての参照先	17
2 Live Upgrade (概要)	19
Live Upgrade の紹介	19
Live Upgrade の処理	20
ブート環境の作成	22
RAID-1 ボリュームファイルシステムを持つブート環境の作成	27
ブート環境のアップグレード	34
ブート環境のアクティブ化	39
元のブート環境へのフォールバック	41
ブート環境の保守	42
3 Live Upgrade (計画)	43
Live Upgrade の要件	43
Live Upgrade のシステム要件	44
Live Upgrade のインストール	44
Live Upgrade のディスク容量の要件	46
RAID-1 ボリューム (ミラー) を作成する場合の Live Upgrade の要件	47
パッケージまたはパッチによるシステムのアップグレード	48
lucreate コマンドを使用したファイルシステムの作成のための指針	49
ファイルシステムのスライスを選択するための指針	50
ルート (/) ファイルシステムのスライスを選択するための指針	50

ミラー化されたファイルシステムのスライスを選択するための指針	51
スワップボリュームのスライスを選択するための指針	52
共有可能なファイルシステムのスライスを選択するための指針	54
新しいブート環境の内容のカスタマイズ	54
ブート環境間でのファイルの同期	55
/etc/lu/synclist ファイルへのファイルの追加	55
ブート環境間での強制的な同期	56
複数のブート環境のブート	57
Live Upgrade のキャラクターインターフェイス	59
4 Live Upgrade を使用したブート環境の作成 (タスク)	61
Live Upgrade のインストールとブート環境の作成 (タスクマップ)	61
Live Upgrade のインストール	62
▼ pkgadd コマンドを使用して Live Upgrade をインストールする方法	63
▼ Oracle Solaris インストールプログラムを使用して Live Upgrade をインストールする 方法	63
Live Upgrade に必要なパッチのインストール	65
新しいブート環境の作成	66
▼ ブート環境をはじめて作成する方法	67
▼ ブート環境を作成しファイルシステムをマージする方法	70
▼ ブート環境を作成してファイルシステムを分割する方法	71
▼ ブート環境を作成しスワップを再構成する方法	73
▼ リストを使用してブート環境を作成しスワップを再構成する方法	75
▼ ブート環境を作成し共有可能ファイルシステムをコピーする方法	78
▼ 別のソースからブート環境を作成する方法	79
▼ フラッシュアーカイブ用の空のブート環境を作成する方法	81
▼ RAID-1 ボリューム (ミラー) を持つブート環境を作成する方法	82
ブート環境の作成とコンテンツのカスタマイズ	88
5 Live Upgrade によるアップグレード (タスク)	91
ブート環境のアップグレード (タスクマップ)	91
ブート環境のアップグレード	92
アップグレードのガイドライン	92
▼ ブート環境のネットワークインストールイメージをアップグレードする方法 ...	94
▼ 複数の CD のネットワークインストールイメージをアップグレードする方法	95

ブート環境のネットワークインストールイメージに対するパッケージの追加またはパッケージの削除	97
ブート環境のネットワークインストールイメージに対するパッチの追加またはパッチの削除	98
ブート環境にインストールされているパッケージの確認	99
JumpStart プロファイルを使用したアップグレード	100
JumpStart キーワード値	105
ブート環境へのフラッシュアーカイブのインストール	108
▼ブート環境にフラッシュアーカイブをインストールする方法	109
▼プロファイルを使用してフラッシュアーカイブをインストールする方法	110
▼プロファイルキーワードを使用してフラッシュアーカイブをインストールする方法	112
ブート環境のアクティブ化	113
ブート環境をアクティブ化するための要件と制限	114
▼ブート環境をアクティブにする方法	115
▼ブート環境をアクティブにしてファイルを同期させる方法	116
x86: GRUB メニューを使ったブート環境のアクティブ化	117
▼x86: GRUB メニューを使ってブート環境をアクティブにする方法	119
6 障害回復:元のブート環境へのフォールバック(タスク)	121
SPARC: 元のブート環境へのフォールバック	122
▼SPARC: 新しいブート環境のアクティブ化に成功した場合のフォールバック	122
▼SPARC: ブート環境のアクティブ化に失敗した場合のフォールバック	122
▼SPARC: DVD、CD、またはネットワークインストールイメージを使って元のブート環境に戻す	123
x86: 元のブート環境へのフォールバック	124
▼x86: 新しいブート環境のアクティブ化に成功した場合の GRUB メニューを使ったフォールバック	124
▼x86: ブート環境のアクティブ化に失敗した場合の GRUB メニューを使ったフォールバック	125
▼x86: ブート環境のアクティブ化に失敗した場合の GRUB メニューと DVD または CD を使ったフォールバック	127
7 Live Upgrade ブート環境の管理(タスク)	131
Live Upgrade の管理の概要	132
ブート環境のための管理作業	132

すべてのブート環境のステータスの表示	132
以前に構成されたブート環境の更新	133
スケジュールされた処理 (作成/アップグレード/コピー) の取り消し	134
ブート環境の比較	135
非アクティブブート環境の削除	135
アクティブブート環境の名前の表示	136
ブート環境の名前の変更	137
ブート環境名に関連付ける説明の作成または変更	138
ブート環境の構成の表示	139
8 非大域ゾーンがインストールされているシステムにおける Oracle Solaris OS のアップグレード	141
Live Upgrade によるアップグレードおよびインストール済みの非大域ゾーン (概要)	142
Oracle Solaris ゾーンと Live Upgrade について	142
非大域ゾーンで Live Upgrade を使用するためのガイドライン (計画)	147
非大域ゾーンが独立したファイルシステム上にある場合にブート環境を作成する	148
非大域ゾーンがインストールされている場合のブート環境の作成とアップグレード (タスク)	149
▼ 非大域ゾーンがシステムにインストールされている場合の Live Upgrade でのアップグレード (タスク)	149
非大域ゾーンがインストールされているシステムのアップグレード (例)	154
非大域ゾーンがシステムにインストールされている場合の Live Upgrade によるアップグレード	154
非大域ゾーンが含まれているブート環境の管理	156
▼ ブート環境の非大域ゾーンのファイルシステムの構成を表示する	156
▼ 非大域ゾーンがインストールされているシステムのブート環境を比較する	157
非大域ゾーンが含まれるシステムで <code>lumount</code> コマンドを使用する	158
9 Live Upgrade の例	159
Live Upgrade によるアップグレードの例	159
Live Upgrade を使用するための準備	160
PreFlightChecker ツールの使用	162
ブート環境を作成する方法	163
非アクティブブート環境をアップグレードする方法	163

ブート環境がブート可能か確認する方法	164
非アクティブブート環境をアクティブにする方法	164
(オプション) ソースブート環境へ戻す方法	164
RAID-1 ボリューム(ミラー)の一方を切り離してアップグレードする例	168
既存のボリュームから Solaris ボリュームマネージャー RAID-1 ボリュームへ移行する 例	172
空のブート環境を作成してフラッシュアーカイブをインストールする例	172
空のブート環境を作成する方法	173
新しいブート環境へフラッシュアーカイブをインストールする方法	174
新しいブート環境をアクティブにする方法	174
 パート II Live Upgrade を使った ZFS ルートプールのアップグレードと ZFS ルートプールへの移 行	175
 10 Live Upgrade と ZFS (概要)	177
Oracle Solaris 10 8/11 リリースの新機能	178
Solaris 10 10/09 リリースの新機能	178
ZFS での Live Upgrade の使用の概要	179
UFS ファイルシステムから ZFS ルートプールへの移行	180
UFS ルート (/) ファイルシステムから ZFS ルートプールへの移行	180
Solaris ボリュームマネージャーボリュームで構成された UFS ファイルシステムの ZFS ルートファイルシステムへの移行	183
ZFS ルートプールからの新規ブート環境の作成	184
同一のルートプール内での新規ブート環境の作成	184
別のルートプール上への新規ブート環境の作成	186
現在稼働中のシステム以外のソースからの新規ブート環境の作成	188
非大域ゾーンがインストールされているシステム上への ZFS ブート環境の作成	188
 11 ZFS の Live Upgrade (計画)	189
Live Upgrade 使用時のシステム要件と制限事項	189
 12 ZFS ルートプールのブート環境の作成	195
UFS ファイルシステムの ZFS ファイルシステムへの移行	195
▼ UFS ファイルシステムを ZFS ファイルシステムに移行する方法	196
同一の ZFS ルートプール内でのブート環境の作成	202

▼ ZFS ブート環境を同一の ZFS ルートプール内に作成する方法	202
新しいルートプール内でのブート環境の作成	207
▼ ブート環境を新規 ZFS ルートプールに作成する方法	207
現在稼働中のシステム以外のソースからのブート環境の作成	212
ZFS ブート環境へのフォールバック	213
13 非大域ゾーンがインストールされている ZFS での Live Upgrade	215
非大域ゾーンがインストールされているシステム上への ZFS ブート環境の作成 (概要と計画)	215
非大域ゾーンがインストールされている UFS ルート (/) ファイルシステムから ZFS ルートプールへの移行 (タスク)	216
▼ 非大域ゾーンのあるシステムで UFS ファイルシステムを ZFS ルートプールに移行する方法	217
パート III 付録	223
A Live Upgrade コマンドリファレンス	225
B トラブルシューティング (タスク)	227
ネットワークインストールの設定に関する問題	227
システムのブートに関する問題	228
メディアからのブート時のエラーメッセージ	228
メディアからのブート時の一般的な問題	229
ネットワークからのブート時のエラーメッセージ	230
ネットワークからのブート時の一般的な問題	234
Oracle Solaris OS の初期インストール	234
▼ x86: IDE ディスクの不良ブロックを検査する方法	235
Oracle Solaris OS のアップグレード	236
アップグレード時のエラーメッセージ	236
アップグレード時の一般的な問題	238
▼ アップグレード失敗後にアップグレードを継続する方法	240
x86: GRUB を使用する場合の Live Upgrade の問題	240
Veritas VxVm の実行中に Live Upgrade を使用してアップグレードするとシステムパニックが発生する	242
x86: 既存のサービスパーティションが存在しないシステムでは、デフォルトで	

サービスパーティションが作成されない	245
▼ ソフトウェアをネットワークインストールイメージまたは Oracle Solaris Operating System DVD からインストールするときにサービスパーティションを含める方法	245
▼ ソフトウェアを Oracle Solaris Software - 1 CD またはネットワークインストールイメージからインストールするときにサービスパーティションを含める方法	246
C その他の SVR4 パッケージ要件 (リファレンス)	247
稼働中の OS に対する変更の防止	247
絶対パスの使用	247
pkgadd -R コマンドの使用	248
\$PKG_INSTALL_ROOT と \$BASEDIR の相違点	248
スクリプト作成のガイドライン	249
ディスクレスクライアントの互換性維持	250
パッケージの検証	250
インストール中およびアップグレード中のユーザー操作の回避	251
ゾーンのパッケージパラメータの設定	252
詳細情報	256
D アップグレード時のパッチアナライザの使用 (タスク)	257
Oracle Solaris Update リリースへのアップグレード	257
▼ analyze_patches を使用する方法	258
 用語集	 261
 索引	 269

はじめに

このドキュメントでは、Oracle Solaris の機能である Live Upgrade を使用して、SPARC および x86 アーキテクチャベースのネットワーク接続されたシステムとネットワーク接続されていないシステムの両方で、Oracle Solaris オペレーティングシステム (OS) をインストールおよびアップグレードする方法について説明します。

このドキュメントには、システムハードウェアや周辺装置を設定する方法は含まれていません。

注 - この Oracle Solaris のリリースでは、SPARC および x86 系列のプロセッサアーキテクチャを使用するシステムをサポートしています。サポートされるシステムは、Oracle Solaris OS: Hardware Compatibility Lists に記載されています。このドキュメントでは、プラットフォームにより実装が異なる場合は、それを特記します。

このドキュメントの x86 に関連する用語については、次を参照してください。

- x86 は、64 ビットおよび 32 ビットの x86 互換製品系列を指します。
- x64 は特に 64 ビット x86 互換 CPU を指します。
- 「32 ビット x86」は、x86 をベースとするシステムに関する 32 ビット特有の情報を指します。

サポートされるシステムについては、[Oracle Solaris OS: Hardware Compatibility Lists](#) を参照してください。

対象読者

このドキュメントは、Oracle Solaris OS のインストールを担当するシステム管理者を対象としています。このドキュメントは、次の 2 種類の情報を提供します。

- ネットワーク接続された環境で複数の Oracle Solaris を管理するエンタープライズシステム管理者のための、高度な Oracle Solaris インストール情報
- Oracle Solaris のアップグレードをときどき実行するシステム管理者のための、基本的な Oracle Solaris インストール情報

関連ドキュメント

次の表に、システム管理者向けドキュメントの一覧を示します。

ヘッダー	ヘッダー
システム要件や計画の概要情報、Oracle Solaris ZFS インストールの概要、ブート、Oracle Solaris ゾーンパーティション分割テクノロジー、または RAID-1 ポリュームの作成などの情報を確認しますか。	『Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画』
DVD または CD メディアから 1 つのシステムをインストールする必要がありますか。Oracle Solaris インストールプログラムは、順を追ってインストールを案内します。	『Oracle Solaris 10 1/13 インストールガイド: 基本インストール』
停止時間をほとんど設けしないで、システムをアップグレードしたり、パッチを適用したりする必要がありますか。Live Upgrade を使用してアップグレードすると、システムの停止時間が削減されます。	『Oracle Solaris 10 1/13 インストールガイド: Live Upgrade とアップグレードの計画』
ネットワークやインターネットを介してセキュリティー保護されたインストールを行う必要がありますか。WAN ブートを使用して、リモートクライアントをインストールします。あるいは、ネットワークインストールイメージからネットワークを介してインストールする必要がありますか。Oracle Solaris インストールプログラムは、順を追ってインストールを案内します。	『Oracle Solaris 10 1/13 インストールガイド: ネットワークベースのインストール』
複数のマシンに Oracle Solaris をインストールする必要がありますか。Oracle Solaris の機能である JumpStart を使用して、インストールを自動化します。	『Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール』
複数のシステムをすばやくインストールしたり、パッチを適用したりする必要がありますか。フラッシュアーカイブソフトウェアを使用してアーカイブを作成し、クローンシステム上に OS のコピーをインストールします。	『Oracle Solaris 10 1/13 インストールガイド: フラッシュアーカイブ (作成とインストール)』
システムのバックアップが必要ですか。	『Oracle Solaris の管理: デバイスとファイルシステム』の第 19 章「UFS ファイルシステムのバックアップと復元 (概要/タスク)」

ヘッダー	ヘッダー
トラブルシューティングに関する情報、既知の問題の一覧、またはこのリリース用のパッチの一覧が必要です。	『Oracle Solaris Release Notes』
システムが Oracle Solaris 上で動作することを確認する必要がありますか。	SPARC: Oracle Solaris Sun のハードウェアプラットフォームガイド
このリリースで追加されたパッケージ、削除されたパッケージ、または変更されたパッケージを確認する必要がありますか。	『Oracle Solaris Package List』
使用しているシステムやデバイスが Solaris SPARC ベースのシステム、x86 ベースのシステム、およびその他のサードパーティーベンダーで動作するかどうかを確認する必要がありますか。	Solaris Hardware Compatibility List (x86 版)

Oracle サポートへのアクセス

Oracle のお客様は、My Oracle Support を通じて電子的なサポートを利用することができます。詳細は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> を参照してください。聴覚に障害をお持ちの場合は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> を参照してください。

表記上の規則

次の表では、このマニュアルで使用される表記上の規則について説明します。

表 P-1 表記上の規則

字体	説明	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	.login ファイルを編集します。 ls -a を使用してすべてのファイルを表示します。 machine_name% you have mail.
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して示します。	machine_name% su Password:
aabbcc123	Placeholder: 実際に使用する特定の名前または値で置き換えます。	ファイルを削除するには、rm filename と入力します。

表 P-1 表記上の規則 (続き)

字体	説明	例
AaBbCc123	書名、新しい単語、および強調する単語を示します。	『ユーザーズガイド』の第6章を参照してください。 キャッシュは、ローカルに格納されるコピーです。 ファイルを保存しないでください。 注: いくつかの強調された項目は、オンラインでは太字で表示されます。

コマンド例のシェルプロンプト

Oracle Solaris OS に含まれるシェルで使用する、UNIX のシステムプロンプトとスーパーユーザープロンプトを次に示します。コマンド例では、シェルプロンプトはコマンドが標準ユーザーまたは特権ユーザーのどちらによって実行されるべきかを示しています。

表 P-2 シェルプロンプト

シェル	プロンプト
Bash シェル、Korn シェル、および Bourne シェル	\$
Bash シェル、Korn シェル、および Bourne シェルのスーパーユーザー	#
C シェル	machine_name%
C シェルのスーパーユーザー	machine_name#

パート I

Live Upgrade を使ったアップグレード

このパートでは、Live Upgrade を使って非アクティブブート環境の作成とアップグレードを行うための概要と手順を説明します。このブート環境は、アクティブブート環境に切り替えることができます。このパートの内容は、UFS ルート (/) ファイルシステムで動作するシステムに当てはまります。ただし、多くのコマンドは ZFS ファイルシステムでも使用できます。

Oracle Solaris インストールの計画についての参照先

このドキュメントでは、Oracle Solaris の機能である Live Upgrade を使用して Oracle Solaris オペレーティングシステムをアップグレードする方法について説明します。このドキュメントでは、Live Upgrade を使用する上で知っておく必要のある情報をすべて提供しますが、開始する前にインストールドキュメントコレクション内の計画ドキュメントを参照しておくで役立ちます。次に示す参照先には、システムをアップグレードする前に役立つ情報が記載されています。

計画とシステム要件についての参照先

『Oracle Solaris 10 8/11 インストールガイド (インストールとアップグレードの計画)』には、システム要件と、ファイルシステムを計画するうえでのガイドラインやアップグレードの計画などの上位計画の概要が記載されています。次の一覧に、この計画ドキュメントの章構成を示します。

計画ガイドの章の説明	リファレンス
この章では、Oracle Solaris OS のインストールやアップグレードを行う前に決定しておく必要のある内容について説明します。ネットワークインストールイメージや DVD メディアをどのようなときに使用するかを判断するための例や、すべての Oracle Solaris インストールプログラムについての説明があります。	『Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画』の第 2 章「Oracle Solaris のインストールおよびアップグレードロードマップ」
この章では、Oracle Solaris OS のインストールやアップグレードに伴うシステム要件について説明します。また、ディスク容量の計画に関しての一般的な指針や、スワップ空間のデフォルトの割り当てについても説明します。アップグレードの制限についても説明します。	『Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画』の第 3 章「システム要件、ガイドライン、およびアップグレード情報」

計画ガイドの章の説明	リファレンス
<p>この章には、システムのインストールやアップグレードに必要な情報の収集に役立つチェックリストが含まれています。ここで提供される情報は、対話式インストールの実行時などに役立ちます。このチェックリストでは、対話式インストールを行うために必要なすべての情報が得られます。</p>	<p>『Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画』の第4章「インストールまたはアップグレードの前の情報の収集」</p>
<p>これらの章では、Oracle Solaris OS のインストールやアップグレードに関連するいくつかの技術の概要を説明します。これらの技術に関連するガイドラインと要件も含まれています。これらの章には、ZFS インストール、ブート、Oracle Solaris ゾーン区分テクノロジー、およびインストール時に作成できる RAID-1 ボリュームについての情報が含まれています。</p>	<p>『Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画』のパート II 「ZFS、ブート、Oracle Solaris ゾーン、および RAID-1 ボリュームに関連するインストールについて」</p>

Live Upgrade (概要)

この章では、Oracle Solaris の機能である Live Upgrade によって使用される処理について説明します。

注- このドキュメントではスライスという用語を使用しますが、一部の Oracle Solaris のドキュメントとプログラムでは、スライスのことを「パーティション」と呼んでいる場合があります。

Live Upgrade の紹介

注- この章では、UFS ファイルシステム用の Live Upgrade について説明します。UFS ファイルシステムの ZFS ルートプールへの移行、または ZFS ルートプールの作成とインストールの概要については、[第 10 章「Live Upgrade と ZFS \(概要\)」](#)を参照してください。

Live Upgrade を使用すると、システムの稼働を続けながらシステムをアップグレードできます。現在のブート環境を動作させたまま、ブート環境のコピーを作成し、それをアップグレードできます。アップグレードする代わりに、フラッシュアーカイブをブート環境にインストールすることもできます。環境をアップグレードしても、アーカイブをインストールしても、元のシステム構成は影響を受けずに支障なく機能します。準備ができたところでシステムをリブートすると、新しいブート環境がアクティブになります。障害が発生した場合は、リブートするだけで元のブート環境に戻ることができます。このように切り替えが可能なので、テストや評価処理のためにサービスを停止する必要がなくなります。

Live Upgrade を使用すると、現在動作しているシステムに影響を与えずに、ブート環境を複製できます。次のような作業を行うことができます。

- システムをアップグレードします。

- 現在のブート環境のディスク構成を、新しいブート環境のディスク構成（ファイルシステムのタイプ、サイズ、および配置）に変更します。
- 異なるイメージを持つ複数のブート環境を保守します。たとえば、現在のパッチを持つブート環境を作成すると同時に、Update リリースを持つ別のブート環境を作成できます。

ヒント - Live Upgrade の PreFlightChecker ツールを使用すると、Live Upgrade の動作の失敗を招くことがあるシステム構成の問題を検出できます。このツールは、`lucreate` コマンドを起動してブート環境を作成する前に実行しておく必要があります。Live Upgrade PreFlightChecker ツールの詳細については、[lupc\(1M\)](#) のマニュアルページを参照してください。

PreFlightChecker ツールの使用例については、[162 ページの「PreFlightChecker ツールの使用」](#)を参照してください。

Live Upgrade を使用する前に、基本的なシステム管理について理解しておく必要があります。ファイルシステムの管理、マウント、ブート、スワップの管理など、システム管理タスクに関する基本的な情報については、『[Oracle Solaris の管理: デバイスとファイルシステム](#)』を参照してください。

Live Upgrade の処理

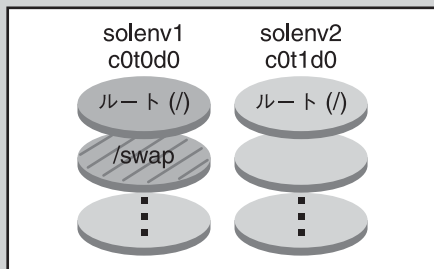
次に、現在のブート環境のコピーを作成してそのコピーをアップグレードし、アクティブなブート環境になるように切り替えるタスクの概要を示します。元のブート環境に切り替えるフォールバックの手順についても説明します。[図 2-1](#) に、この Live Upgrade の処理全体を示します。

図 2-1 Live Upgrade の処理

Solaris Live Upgrade の実行手順

① ブート環境を作成する。

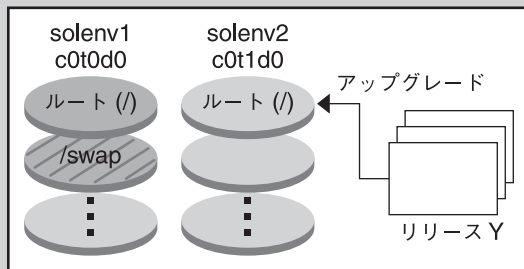
```
# lucreate -c solenv1 \
-m /dev/dsk/c0t1d0s0:ufs \
-n solenv2
```



② 非アクティブブート環境をアップグレードする。

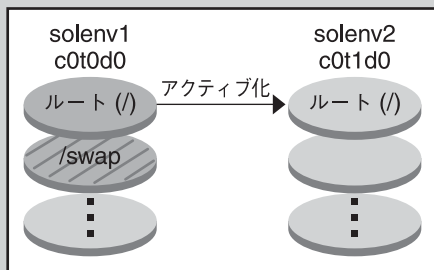
標準アップグレードの場合

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



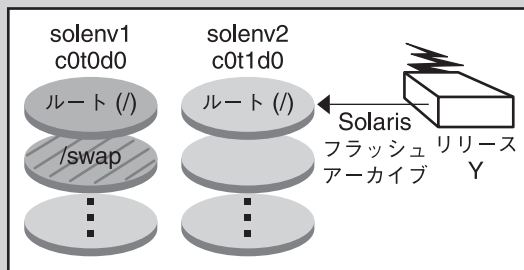
③ リブートして非アクティブブート環境をアクティブ化する。

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



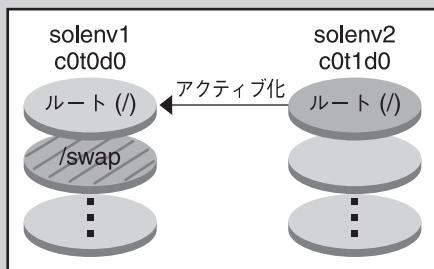
Solaris フラッシュアーカイブの場合

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



④ (任意) 元のブート環境へフォールバックする。

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



⑤ (任意) 非アクティブ環境を削除する。

```
# ludelete solenv2
```

次のセクションで、Live Upgrade の処理について説明します。

1. 物理スライスまたは論理ボリューム上での新しいブート環境の作成
 - [22 ページの「ブート環境の作成」](#)
 - [27 ページの「RAID-1 ボリュームファイルシステムを持つブート環境の作成」](#)
2. [34 ページの「ブート環境のアップグレード」](#)
3. [39 ページの「ブート環境のアクティブ化」](#)
4. [41 ページの「元のブート環境へのフォールバック」](#)

ブート環境の作成

ブート環境を作成すると、クリティカルファイルシステムをアクティブなブート環境から新しいブート環境にコピーできます。必要であれば、ディスクを編成し直して、ファイルシステムをカスタマイズし、クリティカルファイルシステムを新しいブート環境にコピーします。

ファイルシステムのタイプ

Live Upgrade では、クリティカルファイルシステムと共有可能ファイルシステムの2種類のファイルシステムを区別します。次の表に、これらのファイルシステムのタイプを示します。

ファイルシステムのタイプ	説明	例と詳細
クリティカルファイルシステム	クリティカルファイルシステムは、Oracle Solaris OS に必須のファイルシステムです。これらのファイルシステムは、アクティブなブート環境と非アクティブなブート環境の <code>vfstab</code> において別々のマウントポイントを持ちます。これらのファイルシステムは、必ずソースブート環境から非アクティブブート環境にコピーされます。クリティカルファイルシステムのことを「共有不能」と呼ぶこともあります。	<code>root (/)</code> 、 <code>/usr</code> 、 <code>/var</code> 、 <code>/opt</code> などがクリティカルファイルシステムの例です。
共有可能ファイルシステム	共有可能なファイルシステムとは、 <code>/export</code> のように、アクティブなブート環境と非アクティブなブート環境の両方の <code>vfstab</code> において同じマウントポイントを持つユーザー定義ファイルのことです。したがって、アクティブなブート環境内の共有ファイルを更新すると、非アクティブなブート環境のデータも更新されます。新しいブート環境を作成するとき、共有可能なファイルシステムはデフォルトで共有されます。しかし、コピー先のスライスを指定した場合、そのファイルシステムは(共有されずに)コピーされます。	たとえば、 <code>/export</code> が共有可能ファイルシステムの例です。 共有可能なファイルシステムについての詳細は、 54 ページの「共有可能なファイルシステムのスライスを選択するための指針」 を参照してください。

ファイルシステムのタイプ	説明	例と詳細
スワップ	<ul style="list-style-type: none"> ■ UFS ファイルシステムでは、スワップは特殊な共有可能ボリュームです。ほかの共有可能なファイルシステムと同様に、すべてのスワップスライスはデフォルトで共有されます。しかし、スワップ用のディレクトリを指定した場合、スワップスライスは(共有されずに)コピーされます。 ■ ZFS ファイルシステムでは、スワップボリュームとダンプボリュームがプール内で共有されます。 	<ul style="list-style-type: none"> ■ UFS ファイルシステムのスワップを再構成する手順については、73 ページの「ブート環境を作成しスワップを再構成する方法」を参照してください。 ■ ZFS ルートプールのスワップについては、189 ページの「Live Upgrade 使用時のシステム要件と制限事項」を参照してください。

ファイルシステム上の RAID-1 ボリュームの作成

Live Upgrade では、ファイルシステム上に RAID-1 ボリューム (ミラー) を持つブート環境を作成できます。概要については、[27 ページの「RAID-1 ボリュームファイルシステムを持つブート環境の作成」](#)を参照してください。

ファイルシステムのコピー

新しいブート環境を作成するには、まず、クリティカルファイルシステムをコピーできる未使用のスライスが存在することを確認します。スライスが使用できないかあるいは最小限の要件を満たしていない場合は、新しいスライスをフォーマットする必要があります。

スライスを定義した後、ファイルシステムをディレクトリにコピーする前に、新しいブート環境上のファイルシステムを再構成できます。ファイルシステムを分割およびマージすることによって `vfstab` を簡単に編集でき、ファイルシステムを再構成することができます。ファイルシステムは、同じマウントポイントを指定して親ディレクトリにマージすることも、異なるマウントポイントを指定して親ディレクトリから分割することも可能です。

非アクティブブート環境でファイルシステムを構成した後、自動コピーを開始します。クリティカルファイルシステムは、指定された宛先ディレクトリにコピーされます。共有可能なファイルシステムは(それらの一部をコピーするように指定しない限り)、コピーされずに共有されます。ファイルシステムをアクティブなブート環境から非アクティブなブート環境にコピーする時、ファイルは新しいディレクトリにコピーされるので、アクティブなブート環境は変更されません。

ファイルシステムの分割やマージの手順	<ul style="list-style-type: none"> ■ 70 ページの「ブート環境を作成しファイルシステムをマージする方法」 ■ 71 ページの「ブート環境を作成してファイルシステムを分割する方法」
--------------------	--

RAID-1 ボリュームファイルシステムを持つ
ブート環境の作成の概要

27 ページの「RAID-1 ボリュームファイルシス
テムを持つブート環境の作成」

新しいブート環境の作成の例

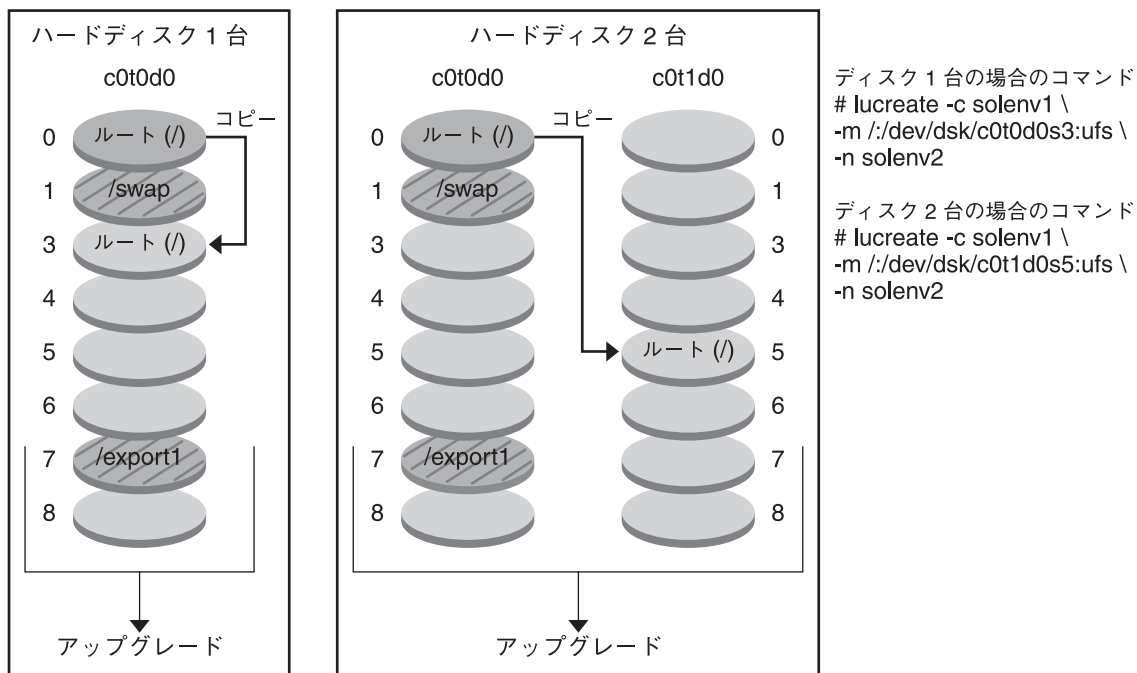
UFS ファイルシステムについては、このセクションの図で、新しいブート環境を作成するためのさまざまな方法について説明します。

ZFS ファイルシステムについては、[第 10 章「Live Upgrade と ZFS \(概要\)」](#)を参照してください。

次の図は、クリティカルファイルシステムのルート (/) をディスク上の別のスライスにコピーして、新しいブート環境を作成する様子を示しています。アクティブなブート環境は、既存のスライス上にルート (/) ファイルシステムを持っています。新しいブート環境は、新しいスライス上にルート (/) ファイルシステムとまったく同じ複製を持ちます。/swap ボリュームと /export/home ファイルシステムは、アクティブなブート環境と非アクティブなブート環境で共有されます。

図2-2 非アクティブなブート環境の作成-ルート(/)ファイルシステムのコピー

ブート環境の作成 -
スライスへのルート (h) ファイルシステムのコピー

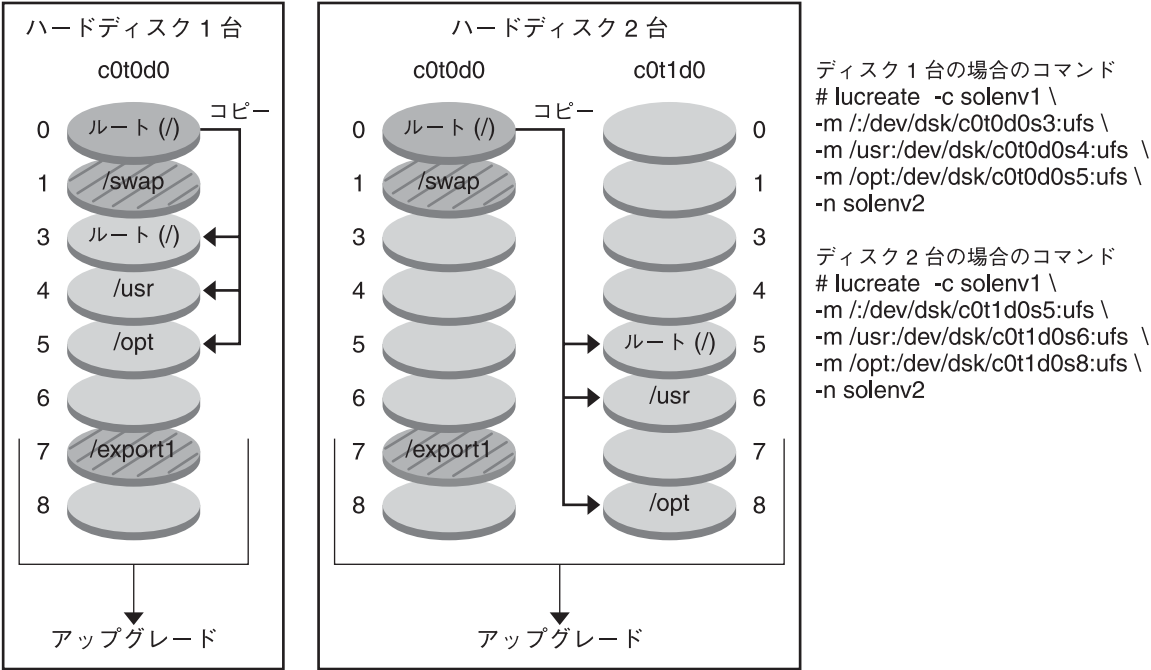


- ☐ 現在のリリース X
クリティカルなファイルシステムのルート (/)
- ☐ 非アクティブなリリース X
クリティカルなファイルシステムのルート (/)
- ☒ 共有ファイルシステム

次の図は、新しいブート環境を作成するために、分割され、ディスク上の複数のスライスにコピーされたクリティカルファイルシステムを示しています。アクティブなブート環境は、既存のスライス上にルート (/) ファイルシステムを持っています。このスライスでは、ルート (/) ファイルシステム内に、`/usr`、`/var`、および `/opt` ディレクトリがあります。新しいブート環境では、ルート (/) ファイルシステムは分割され、`/usr` と `/opt` は別のスライスに配置されています。`/swap` ボリュームと `/export/home` ファイルシステムは、両方のブート環境で共有されます。

図 2-3 非アクティブなブート環境の作成 - ファイルシステムの分割

ブート環境の作成 - ファイルシステムの分割

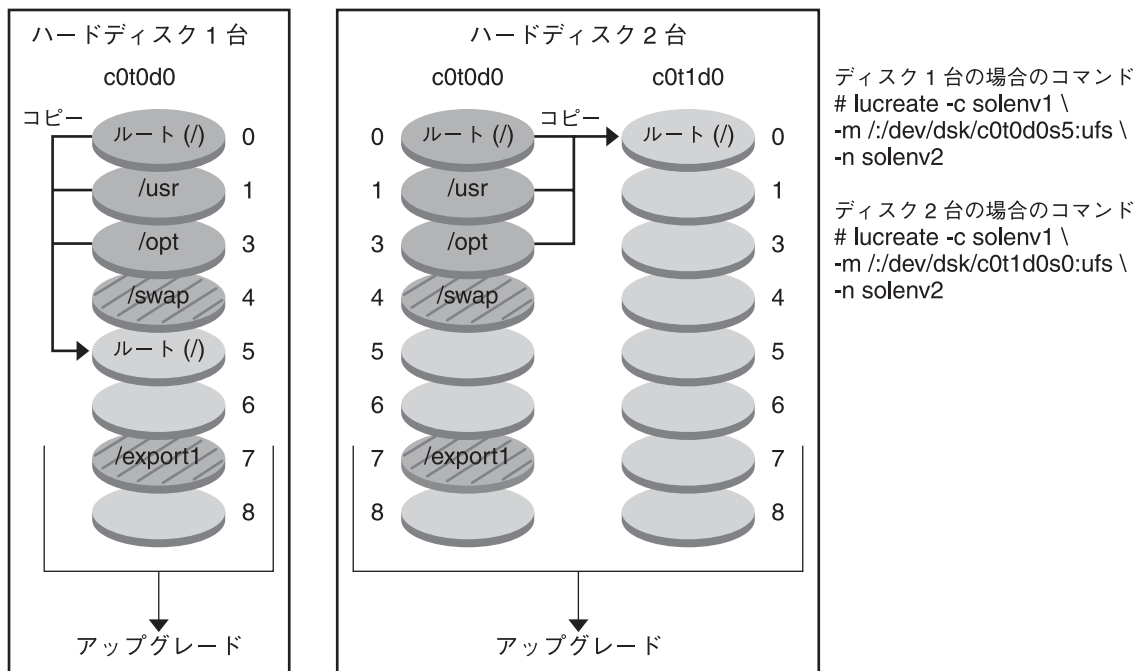


- 現在のリリース X
クリティカルなファイルシステムのルート (/)
- 非アクティブなリリース X
クリティカルなファイルシステムのルート (/) /usr /opt
- 共有ファイルシステム

次の図は、新しいブート環境を作成するために、マージされ、ディスク上の複数のスライスにコピーされたクリティカルファイルシステムを示しています。アクティブなブート環境には、ルート (/) ファイルシステム、/usr、/var、/opt があり、各ファイルシステムは独自のスライス上に配置されています。新しいブート環境では、/usr と /opt は 1 つのスライス上のルート (/) ファイルシステムにマージされます。/swap ボリュームと /export/home ファイルシステムは、両方のブート環境で共有されます。

図 2-4 非アクティブなブート環境の作成 - ファイルシステムのマージ

ブート環境の作成 - ファイルシステムのマージ



- 現在のリリース X
クリティカルなファイルシステムのルート (/) /usr /opt
- 非アクティブなリリース Y
クリティカルなファイルシステムのルート (/)
- ▨ 共有ファイルシステム

RAID-1 ボリュームファイルシステムを持つブート環境の作成

Live Upgrade は、Solaris Volume Manager テクノロジを使用して、RAID-1 ボリュームにカプセル化されたファイルシステムを持つブート環境を作成します。Solaris ボリュームマネージャーでは、ボリュームを使って確実にディスクやデータを管理できます。Solaris ボリュームマネージャーでは、連結、ストライプ、その他の複雑な構成が可能です。Live Upgrade では、これらのタスクの一部 (ルート (/) ファイルシステムの RAID-1 ボリュームの作成など) を実行できます。

ボリュームを使用すると、複数のディスクにまたがるディスクスライスをグループ化して、OS で単一のディスクとして扱われるようにできます。Live Upgrade で作成

できるのは、RAID-1 ボリューム (ミラー) 内に単一スライスの連結を持つルート (/) ファイルシステムのブート環境だけです。これは、ブート用のスライスを 1 つだけ選択するようにブート PROM が制限されているためです。

Live Upgrade でのボリュームの管理

ブート環境を作成するとき、Live Upgrade を使って次のタスクを行うことができます。

- 単一スライスの連結 (サブミラー) を RAID-1 ボリューム (ミラー) から切り離します。必要な場合は、内容を保持して新しいブート環境の内容にすることができます。内容はコピーされないため、新しいブート環境を短時間で作成できます。ミラーから切り離されたサブミラーは、元のミラーの一部ではなくなります。サブミラーに対する読み取りや書き込みがミラーを介して実行されることはなくなります。
- ミラーを含んだブート環境を作成します。
- 新しく作成したミラーに単一スライスの連結を 3 つまで接続します。

lucreate コマンドの `-m` オプションを使って、新しいブート環境に対してミラーの作成、サブミラーの切り離し、およびサブミラーの接続を行うことができます。

注-現在のシステム上に VxVM ボリュームが構成されている場合は、lucreate コマンドを使用して新しいブート環境を作成できます。新しいブート環境にデータをコピーすると、Veritas ファイルシステム構成が失われ、新しいブート環境に UFS ファイルシステムが作成されます。

詳細は、次のリソースを参照してください。

- 詳細な手順については、[82 ページの「RAID-1 ボリューム \(ミラー\) を持つブート環境を作成する方法」](#)を参照してください。
- インストール中の RAID-1 ボリューム作成の概要については、『[Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画](#)』の第 8 章「インストール時の RAID-1 ボリューム (ミラー) の作成 (概要)」を参照してください。
- Live Upgrade を使用する場合にサポートされない Solaris Volume Manager のほかの複雑な構成についての詳細は、『[Solaris Volume Manager 管理ガイド](#)』の第 2 章「ストレージ管理の概念」を参照してください。

Solaris Volume Manager のタスクと Live Upgrade の対応

Live Upgrade では、Solaris Volume Manager のタスクの一部を管理します。次の表に、Live Upgrade で管理できる Solaris Volume Manager コンポーネントを示します。

表 2-1 ポリウムクラス

用語	説明
連結	RAID-0 ポリウム。複数のスライスが連結された方式では、利用可能な最初のスライスがいっぱいになるまでそのスライスにデータが書き込まれます。そのスライスがいっぱいになると次のスライスに連続してデータが書き込まれます。ミラーに含まれている場合を除き、連結にはデータの冗長性はありません。
ミラー	RAID-1 ポリウム。「RAID-1 ポリウム」を参照してください。
RAID-1 ポリウム	同じデータのコピーを複数保持しているポリウム。RAID-1 ポリウムは「ミラー」と呼ばれることもあります。RAID-1 ポリウムは、サブミラーと呼ばれる 1 つまたは複数の RAID-0 ポリウムから構成されます。
RAID-0 ポリウム	ストライプ方式または連結方式のポリウム。これらのコンポーネントは、サブミラーとも呼ばれます。ストライプや連結は、ミラーを構築する基本構成ブロックです。
状態データベース	状態データベースでは、Solaris ポリウムマネージャー構成の状態に関する情報がディスクに保存されます。状態データベースは、複製された複数のデータベースコピーの集まりです。各コピーは「状態データベースの複製」と呼ばれます。状態データベースは、既知の状態データベースの複製の格納場所とステータスをすべて記録しています。
状態データベースの複製	状態データベースのコピー。複製により、データベース内のデータの有効性が保証されます。
サブミラー	「RAID-0 ポリウム」を参照してください。
ポリウム	システムで単一の論理デバイスとして扱われる、物理スライスやポリウムの集まり。アプリケーションやファイルシステムから見ると、ポリウムは物理ディスクと同じように機能します。一部のコマンド行ユーティリティーでは、ポリウムはメタデバイスと呼ばれます。

Live Upgrade を使用して RAID-1 ポリウムを作成する例

このセクションの例では、新しいブート環境の RAID-1 ポリウムを作成するためのコマンド構文を示します。

2つの物理ディスク上に RAID-1 ポリウムを作成する

次の図は、2つの物理ディスク上に作成された RAID-1 ポリウム(ミラー)を持つ新しいブート環境を示しています。この新しいブート環境とミラーは、次のコマンドで作成されました。

```
# 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
```

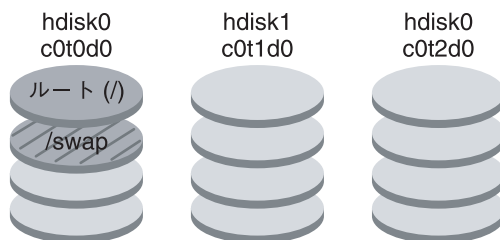
このコマンドは、次のようなタスクを実行します。

- 新しいブート環境 `second_disk` を作成する。
- ミラー `d30` を作成し、UFS ファイルシステムを構成する。
- 各物理ディスクのスライス 0 に単一デバイスの連結を作成する。これらの連結に `d31` および `d32` という名前を付ける。
- これら 2 つの連結をミラー `d30` に追加する。
- ルート (`/`) ファイルシステムをミラーにコピーする。
- 各物理ディスクのスライス 1 に、スワップ用のファイルシステムを構成する。

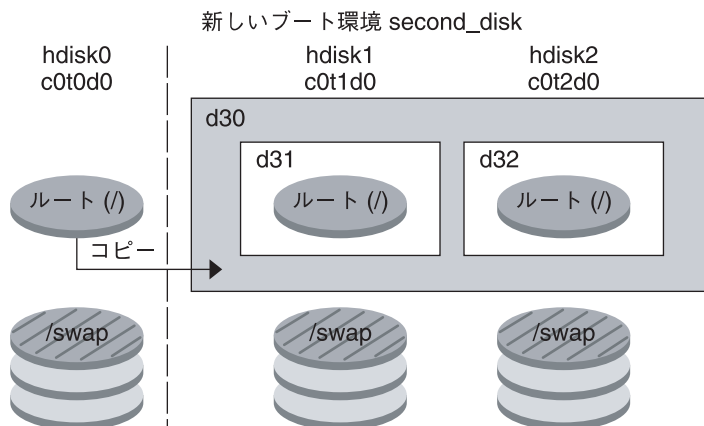
図 2-5 ブート環境の作成とミラーの作成

ミラーを使った新しいブート環境の作成

3 つの物理ディスクからなる元のシステム



コマンド: `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`



d30 – RAID-1 ボリューム (ミラー)

d31 – 単一スライスの連結 (サブミラー)

d32 – 単一スライスの連結 (サブミラー)

ブート環境の作成と既存のサブミラーの使用

次の図は、RAID-1 ボリューム (ミラー) を持つ新しいブート環境を示しています。この新しいブート環境とミラーは、次のコマンドで作成されました。

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

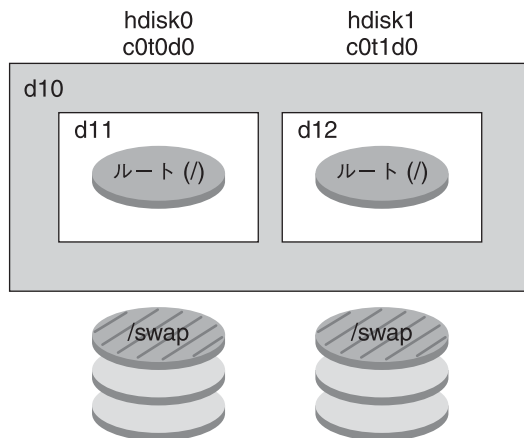
このコマンドは、次のようなタスクを実行します。

- 新しいブート環境 `second_disk` を作成する。

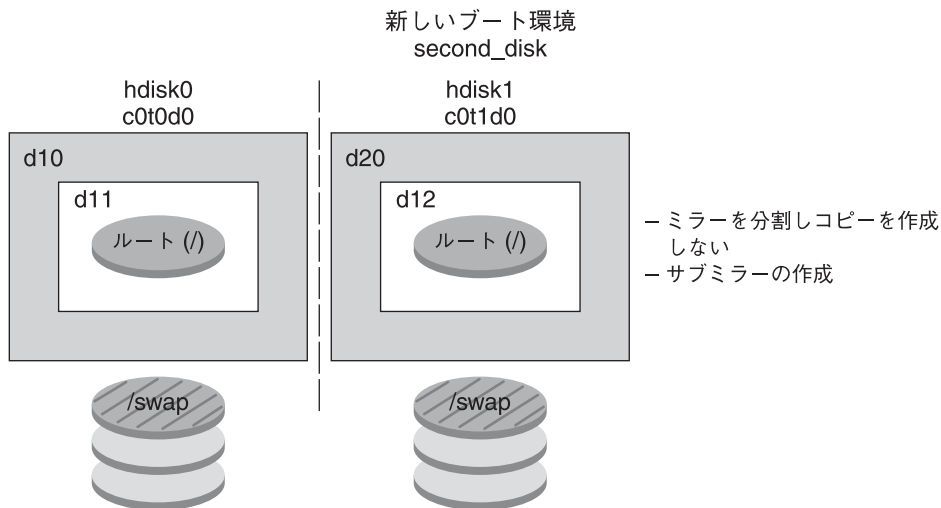
- ミラー d10 を解除し、連結 d12 を切り離します。
- 連結 d12 の内容を保持します。ファイルシステムのコピーは行われません。
- 新しいミラー d20 を作成した結果、2つの一方向ミラー d10 および d20 が作成されます。
- 連結 d12 をミラー d20 に接続します。

図 2-6 ブート環境の作成と既存のサブミラーの使用
新しいブート環境の作成と既存のサブミラーの流用

2つの物理ディスクからなる元のシステム



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



d10 – RAID-1 ボリューム (ミラー)
d11 – 単一スライスの連結 (サブミラー)
d12 – 単一スライスの連結 (サブミラー)
d20 – 新しい RAID-1 ボリューム (ミラー)

ブート環境のアップグレード

ブート環境の作成が完了したら、そのブート環境をアップグレードできます。アップグレードの一部として、ブート環境には任意のファイルシステムの RAID-1 ボリューム(ミラー)を含めたり、非大域ゾーンをインストールしたりできます。アップグレードを行なっても、アクティブなブート環境内のファイルには影響ありません。準備ができたところでこの新しいブート環境をアクティブにし、このブート環境を現行のブート環境とします。

注 - **Oracle Solaris 10 9/10** リリース以降、アップグレード処理は自動登録の影響を受けます。37 ページの「[Live Upgrade に対する自動登録の影響](#)」を参照してください。

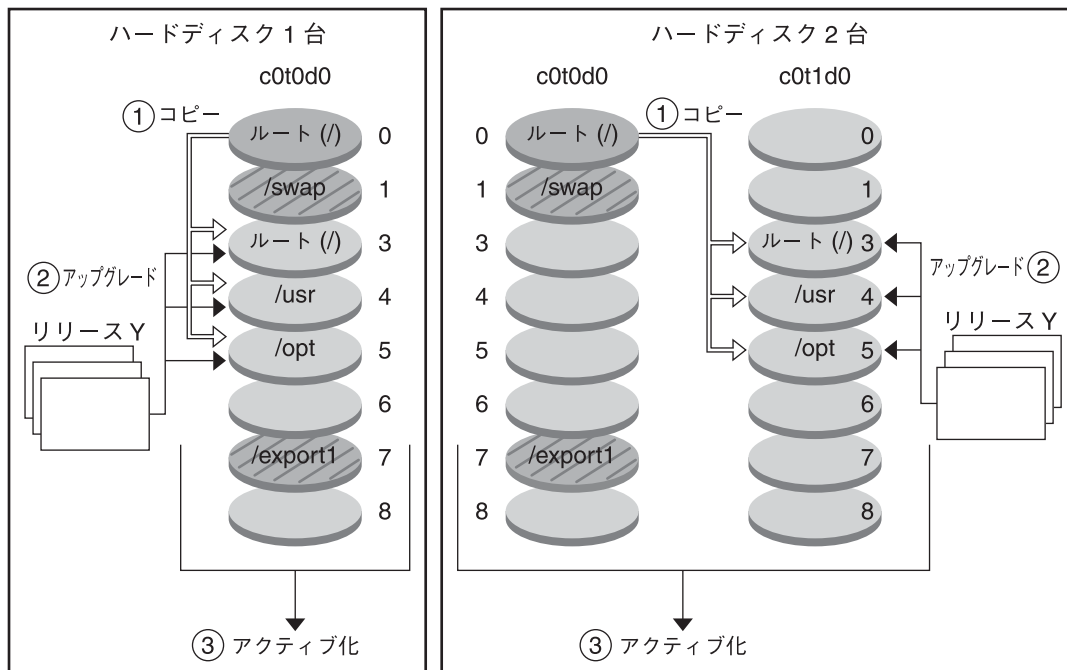
詳細は、次のリソースを参照してください。

- UFS ファイルシステムのブート環境をアップグレードする手順については、第 5 章「[Live Upgrade によるアップグレード\(タスク\)](#)」を参照してください。
- UFS ファイルシステム用の RAID-1 ボリュームファイルシステムを持つブート環境をアップグレードする例については、168 ページの「[RAID-1 ボリューム\(ミラー\)の一方を切り離してアップグレードする例](#)」を参照してください。
- UFS ファイルシステムの非大域ゾーンのアップグレード手順については、第 8 章「[非大域ゾーンがインストールされているシステムにおける Oracle Solaris OS のアップグレード](#)」を参照してください。
- ZFS ファイルシステムのアップグレードまたは ZFS ファイルシステムへの移行については、第 10 章「[Live Upgrade と ZFS \(概要\)](#)」を参照してください。

次の図は、非アクティブなブート環境のアップグレードを示しています。

図 2-7 非アクティブなブート環境のアップグレード

ブート環境のアップグレード



- 現在のリリース X
クリティカルなファイルシステムのルート (/)
- 非アクティブなリリース Y
クリティカルなファイルシステムのルート (/)
/usr /opt
- 共有ファイルシステム

① ディスク 1 台の場合のコマンド
`# lucreate -c solenv1 \
 -m /:/dev/dsk/c0t0d0s3:ufs -m /usr:/dev/dsk/c0t0d0s4:ufs \
 -m /opt:/dev/dsk/c0t0d0s5:ufs \
 -n solenv2`

① ディスク 2 台の場合のコマンド
`# 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`

アップグレードする代わりに、フラッシュアーカイブをブート環境にインストールすることもできます。フラッシュアーカイブインストール機能を使用すると、Oracle Solaris OS の単一の参照用インストールを 1 台のシステム上に作成できます。このシステムはマスターシステムと呼ばれます。続いて、クローンシステムと呼ばれる多数のシステム上にこのインストールを複製できます。この場合、非アクティブなブート環境はクローンシステムです。フラッシュアーカイブをシステムにインス

ツールするとき、初期インストールの場合と同じように、アーカイブは既存のブート環境にあるすべてのファイルを置き換えます。

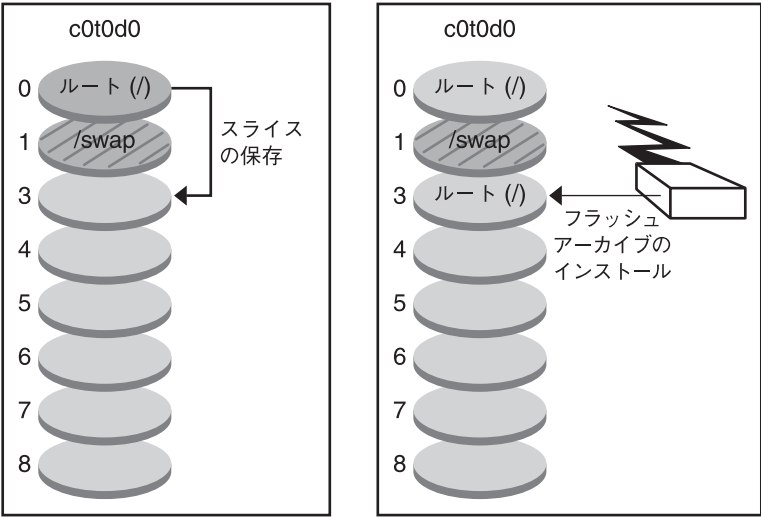
フラッシュアーカイブのインストール手順については、108 ページの「ブート環境へのフラッシュアーカイブのインストール」を参照してください。

次の図に、非アクティブなブート環境におけるフラッシュアーカイブのインストールを示します。図 2-8 は、1 台のハードディスクを持つシステムを示しています。図 2-9 は、2 台のハードディスクを持つシステムを示しています。

図 2-8 単一ディスクへのフラッシュアーカイブのインストール

Solaris フラッシュアーカイブのインストール – ハードディスク 1 台

- ① 空のブート環境の作成
- ② フラッシュアーカイブをインストールすることによるアップグレード



- 現在のリリース X
クリティカルなファイルシステムのルート (/)
- 非アクティブなリリース Y
クリティカルなファイルシステムのルート (/)
/usr /opt
- ▨

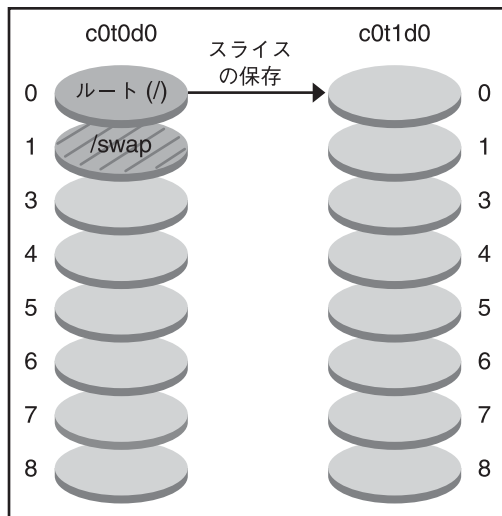
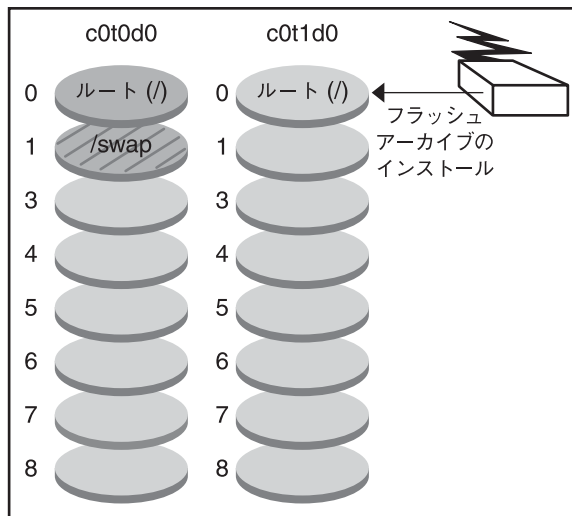
共有ファイルシステム
- コマンド
lucreate -s - \
-m /:/dev/dsk/c0t0d0s3:ufs -n solenv2

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

図 2-9 2つのディスクへのフラッシュアーカイブのインストール

Solaris フラッシュアーカイブのインストーラー – ハードディスク 2 台

空のブート環境の作成

フラッシュアーカイブを
インストールすることによるアップグレード

- 現在のリリース X
クリティカルなファイルシステムのルート (/)
- 非アクティブなリリース X
クリティカルなファイルシステムのルート (/)
- ▣ 共有ファイルシステム

コマンド

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

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

Live Upgrade に対する自動登録の影響

Oracle Solaris 10 9/10 リリース以降、アップグレード処理は自動登録の影響を受けません。

自動登録について

システムをインストールまたはアップグレードすると、システムの構成データは、既存のサービスタグ技術によってリブート時に自動的にオラクル製品登録システムに伝達されます。システムに関するこのサービスタグデータは、オラクルの顧客向けサポートとサービスの向上などに役立てられます。この同じ構成データを使用して、システム独自の目録を作成および管理することができます。

自動登録が Live Upgrade に影響を与える状況

システムを明示的に、以前のリリースから Oracle Solaris 10 9/10 リリースまたはそれ以降のリリースにアップグレードしようとしている場合を除き、自動登録によって Live Upgrade の処理は変更されません。

Live Upgrade の次の手順は、自動登録によって変更されません。

- フラッシュアーカイブのインストール
- パッチやパッケージの追加または削除
- プロファイルのテスト
- パッケージの整合性の確認

システムを以前のリリースから Oracle Solaris 10 9/10 リリースまたはそれ以降のリリースにアップグレードする場合のみ、自動登録の構成ファイルを作成する必要があります。そしてシステムをアップグレードするときには、`luupgrade -u` コマンドで `-k` オプションを使用して、この構成ファイルを指定する必要があります。

▼ アップグレード中に自動登録の情報を提供する方法

以前のリリースから Oracle Solaris 10 9/10 リリースまたはそれ以降のリリースにアップグレードする場合のみ、この手順を使用して、必要とされる自動登録の情報をアップグレード中に提供します。

- 1 サポート資格を含み、必要に応じてプロキシ情報も含めた構成ファイルを作成します。

このファイルは、キーワードと値のペアから成るリストの書式にします。ファイルには、この書式で次のキーワードと値を含めます。

```
http_proxy=Proxy-Server-Host-Name
http_proxy_port=Proxy-Server-Port-Number
http_proxy_user=HTTP-Proxy-User-Name
http_proxy_pw=HTTP-Proxy-Password
oracle_user=My-Oracle-Support-User-Name
oracle_pw=My-Oracle-Support-Password
```

次の書式ルールに注意してください。

- パスワードは暗号化テキストではなく、平文テキストにする必要があります。
- キーワードの順序は重要ではありません。
- 値を指定しない場合は、キーワードを完全に省略できます。または、キーワードを保持して値を空白のままにすることもできます。

注- サポート資格を省略すると、登録は匿名になります。

- 入力する値にスペースを含める必要がある場合を除き、構成ファイルのスペースは問題になりません。http_proxy_user と http_proxy_pw の値のみ、値の中にスペースを含むことができます。
- oracle_pw の値にスペースを含めてはいけません。

次にサンプルファイルの例を示します。

```
http_proxy= webcache.central.example.COM
http_proxy_port=8080
http_proxy_user=webuser
http_proxy_pw=secret1
oracle_user=joe.smith@example.com
oracle_pw=csdfl2442IJS
```

- 2 ファイルを保存します。
- 3 そのアップグレードに必要な **luupgrade** コマンドのその他の標準オプションをすべて含めて、**luupgrade -u -k /path/filename** コマンドを実行します。

▼ アップグレード中に自動登録を無効にする方法

- 1 構成ファイルを作成するか、作成した既存の構成ファイルの内容を修正して、ファイルに次の行のみが含まれるようにします。

```
autoreg=disable
```

- 2 ファイルを保存します。
- 3 そのアップグレードに必要な **luupgrade** コマンドのその他の標準オプションをすべて含めて、**luupgrade -u -k /path/filename** コマンドを実行します。
- 4 (省略可能) **Live Upgrade** が完了し、システムがリブートするとき、次のようにして自動登録機能が無効になっていることを確認します。

```
# /opt/ocm/ccr/bin/emCCR status
Oracle Configuration Manager - Release: 10.3.6.0.1 - Production
Copyright (c) 2005, 2011, Oracle and/or its affiliates. All rights reserved.
-----
Log Directory          /opt/ocm/config_home/ccr/log
Collector Mode          Disconnected
```

ブート環境のアクティブ化

新しいブート環境に切り替えてアクティブにする準備ができたなら、新しいブート環境を簡単にアクティブにしてリブートできます。新たに作成したブート環境を初めてブートするとき、ブート環境間でファイルの同期がとられます。ここでいう「同期」とは、いくつかのシステムファイルやディレクトリを、直前にアクティブだったブート環境から、ブート中のブート環境へコピーすることです。システムを

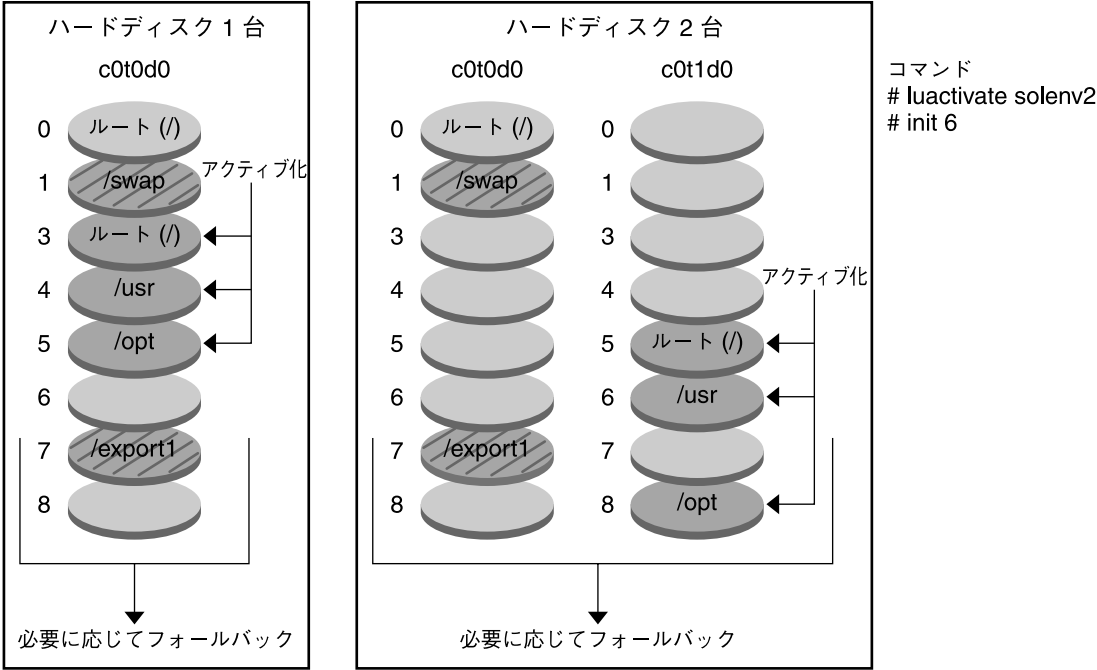
リブートすると、非アクティブなブート環境にインストールした構成がアクティブになります。この時点で、元のブート環境は非アクティブブート環境となります。

ブート環境をアクティブにする手順については、113 ページの「ブート環境のアクティブ化」を参照してください。アクティブなブート環境と非アクティブなブート環境の同期については、55 ページの「ブート環境間でのファイルの同期」を参照してください。

次の図に、リブート後の非アクティブなブート環境からアクティブなブート環境への切り替えを示します。

図 2-10 非アクティブなブート環境のアクティブ化

ブート環境のアクティブ化



- 現在のリリース Y
クリティカルなファイルシステムのルート (/) /usr /opt
- 非アクティブなリリース X
クリティカルなファイルシステムのルート (/)
- 共有ファイルシステム

元のブート環境へのフォールバック

問題が発生する場合は、アクティブ化とリブートを行なって元のブート環境にすぐにフォールバックできます。システムをリブートするのに、元のブート環境をバックアップして復元するよりも、フォールバックの方がはるかに時間がかかりません。ブートに失敗した新しいブート環境は保存されるので、障害を解析できます。新しいブート環境をアクティブにするために `luactivate` によって使用されたブート環境だけにフォールバックできます。

以前のブート環境にフォールバックできる方法を次の表で説明します。

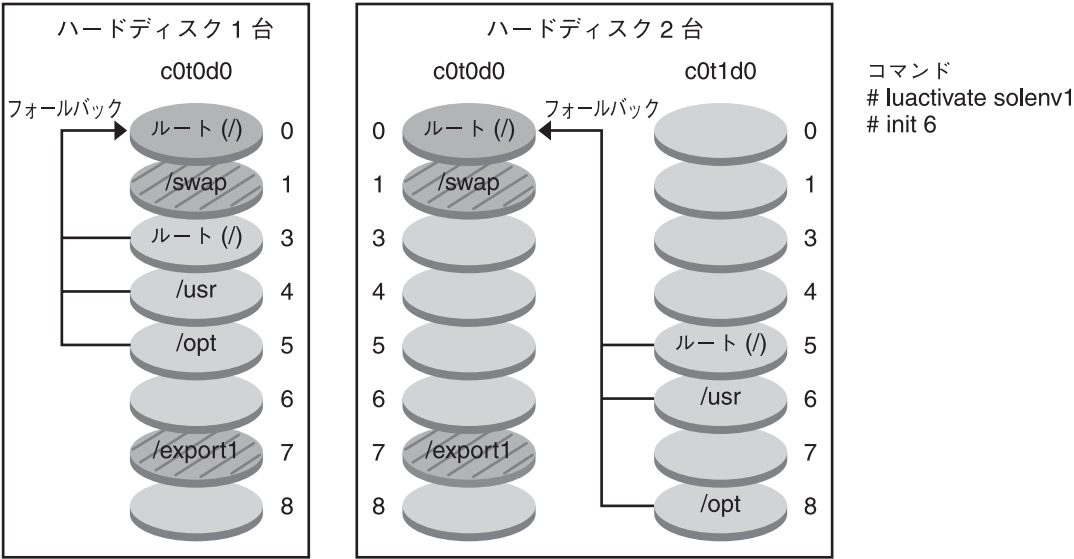
問題	対処方法
新しいブート環境は正常にブートしたが、結果に満足できない。	<p><code>luactivate</code> コマンドに以前のブート環境の名前を指定して実行し、リブートします。</p> <p>x86 のみ - Solaris 10 1/06 以降のリリースでは、GRUB メニューにある元のブート環境を選択して戻すことができます。元のブート環境と新しいブート環境は、GRUB ソフトウェアに基づいている必要があります。GRUB メニューからブートすると、古いブート環境と新しいブート環境の間でファイルは同期されません。ファイルの同期の詳細については、56 ページの「ブート環境間での強制的な同期」を参照してください。</p>
新しいブート環境がブートしない。	戻すブート環境をシングルユーザーモードでブートし、 <code>luactivate</code> コマンドを実行し、リブートします。
シングルユーザーモードでブートできない。	<p>次のいずれかの操作を実行します。</p> <ul style="list-style-type: none"> ■ DVD/CD メディアまたはネットインストールイメージからブートします ■ 戻すブート環境上のルート (/) ファイルシステムをマウントします ■ <code>luactivate</code> コマンドを実行し、リブートします

フォールバックの手順については、[第 6 章「障害回復: 元のブート環境へのフォールバック \(タスク\)」](#)を参照してください。

次の図に、リブートしてフォールバックしたときの切り替えを示します。

図 2-11 元のブート環境へのフォールバック

元のブート環境へのフォールバック



- 現在のリリース X
クリティカルなファイルシステムのルート (/)
- 非アクティブなリリース X
クリティカルなファイルシステムのルート (/)
- ▨ 共有ファイルシステム

ブート環境の保守

ブート環境のステータス確認、名前変更、削除など、さまざまな保守作業も行うことができます。保守手順については、[第 7 章「Live Upgrade ブート環境の管理 \(タスク\)」](#)を参照してください。

Live Upgrade (計画)

この章では、Oracle Solaris の機能である Live Upgrade をインストールおよび使用する前に確認しておくガイドラインと要件について説明します。『[Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画](#)』の「アップグレード計画」で、アップグレード全般に関する情報も確認するようにしてください。

注- この章では、UFS ファイルシステム用の Live Upgrade について説明します。ZFS ルートプールへの UFS ファイルシステムの移行、または ZFS ルートプールの作成とインストールの計画については、[第 11 章「ZFS の Live Upgrade \(計画\)」](#)を参照してください。

この章には次のセクションが含まれています。

- [43 ページの「Live Upgrade の要件」](#)
- [48 ページの「パッケージまたはパッチによるシステムのアップグレード」](#)
- [49 ページの「lucreate コマンドを使用したファイルシステムの作成のための指針」](#)
- [50 ページの「ファイルシステムのスライスを選択するための指針」](#)
- [54 ページの「新しいブート環境の内容のカスタマイズ」](#)
- [55 ページの「ブート環境間でのファイルの同期」](#)

Live Upgrade の要件

Live Upgrade のインストールおよび使用を開始する前に、次の要件をよく理解してください。

Live Upgrade のシステム要件

Live Upgrade は Oracle Solaris ソフトウェアに含まれています。現在の OS に Live Upgrade パッケージをインストールする必要があります。Live Upgrade パッケージのリリースは、アップグレード後の OS のリリース番号と一致する必要があります。たとえば、現在使用している OS が Solaris 9 リリースで、Oracle Solaris 10 8/11 リリースにアップグレードする場合、Oracle Solaris 10 8/11 リリースの Live Upgrade パッケージをインストールする必要があります。

次の表に、Live Upgrade でサポートされるリリースを示します。

表 3-1 サポートされる Oracle Solaris リリース

現在のリリース	互換性のあるアップグレードリリース
Solaris 8 OS	Solaris 8、Solaris 9、または任意の Oracle Solaris 10 リリース
Solaris 9 OS	Solaris 9 または任意の Oracle Solaris 10 リリース
Oracle Solaris 10 OS	任意の Oracle Solaris 10 リリース

Live Upgrade のインストール

次のいずれかの方法を使用して Live Upgrade パッケージをインストールできます。

- pkgadd コマンド。Live Upgrade パッケージは SUNWlucfg、SUNWlur、および SUNWluu です。この順序でインストールする必要があります。
- Oracle Solaris Operating System DVD、Oracle Solaris Software - 2 CD、またはネットワークインストールイメージ上にあるインストーラ。

Live Upgrade が正しく動作するためにインストールすることが必要になる場合があるパッチについて、次の情報に注意してください。

Live Upgrade が正しく動作するためには、特定の OS バージョン用の特定のパッチリビジョンのセットがインストールされている必要があります。Live Upgrade をインストールまたは実行する前に、これらのパッチをインストールする必要があります。

x86 のみ - このパッチのセットがインストールされていない場合、Live Upgrade は失敗し、次のエラーメッセージが表示されることがあります。次のエラーメッセージが表示されなくても、必要なパッチがインストールされていない場合があります。Live Upgrade のインストールを試みる前に、My Oracle Support のナレッジドキュメントに記載されたすべてのパッチがインストール済みであることを必ず確認してください。

```
ERROR: Cannot find or is not executable:
</sbin/biosdev>.
ERROR: One or more patches required
by Live Upgrade has not been installed.
```

<http://support.oracle.com> (My Oracle Support) を調べて最新の更新済みパッチリストであることを確認してください。My Oracle Support で、ナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) を検索してください。

My Oracle Support のナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) に記載されているパッチは、いつでも変更されることがあります。これらのパッチにより、Live Upgrade の欠陥が修正される可能性があります。同時に、Live Upgrade が依存するコンポーネントの欠陥も修正される可能性があります。Live Upgrade で問題が発生した場合は、最新の Live Upgrade パッチがインストールされていることを確認してください。

Solaris 8 または Solaris 9 OS を実行している場合は、Live Upgrade インストーラを実行できないことがあります。これらのリリースには、Java 2 Runtime Environment の実行に必要なパッチのセットが含まれていません。Live Upgrade インストーラを実行してパッケージをインストールするには、Java 2 Runtime Environment の推奨パッチクラスタが必要です。Live Upgrade パッケージをインストールするには、pkgadd コマンドを使用します。または、Java 2 Runtime Environment 推奨パッチクラスタをインストールします。パッチクラスタは、<http://support.oracle.com> (My Oracle Support) にあります。

Live Upgrade ソフトウェアのインストール手順については、62 ページの「[Live Upgrade のインストール](#)」を参照してください。

必要なパッケージ

Live Upgrade に問題がある場合は、パッケージが不足している可能性があります。表 3-2 に、各 OS リリースが Live Upgrade を使用するために必要とするパッケージの一覧を示しています。

Oracle Solaris 10 リリースの場合:

- 次のソフトウェアグループには、必要なすべての Live Upgrade パッケージが含まれています。

- 全体ディストリビューションと OEM サポート
- 全体ディストリビューション
- 開発者システムサポート
- エンドユーザーシステムサポート
- 次のソフトウェアグループには、Live Upgrade を使用するために必要なすべてのパッケージが含まれていないことがあります。
 - コアシステムサポート
 - 限定ネットワークシステムサポート

ソフトウェアグループについては、『[Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画](#)』の「ソフトウェアグループごとの推奨ディスク容量」を参照してください。

表 3-2 Live Upgrade に必要なパッケージ

Solaris 8 リリース	Solaris 9 リリース	Oracle Solaris 10 リリース
SUNWadmap	SUNWadmap	SUNWadmap
SUNWadmc	SUNWadmc	SUNWadmlib-sysid
SUNWlibc	SUNWadmfw	SUNWadmr
SUNWbzip	SUNWlibc	SUNWlibc
SUNWgzip	SUNWgzip	Solaris 10 3/05 のみ: SUNWgzip
SUNWj2rt ¹	SUNWj2rt	SUNWj5rt ²

¹ SUNWj2rt パッケージは次の状況でのみ必要になります。

- Live Upgrade インストーラを実行して Live Upgrade パッケージを追加する場合
- アップグレードの際に CD メディアを使用する場合

² SUNWj5rt パッケージは次の状況でのみ必要になります。

- Live Upgrade インストーラを実行して Live Upgrade パッケージを追加する場合
- アップグレードの際に CD メディアを使用する場合

システム上のパッケージを確認するには、次のコマンドを入力します。

% **pkginfo** *package-name*

Live Upgrade のディスク容量の要件

アップグレードのためのディスク容量の要件については、『[Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画](#)』の第 3 章「システム要件、ガイドライン、およびアップグレード情報」を参照してください。

ブート環境の作成に必要なファイルシステムのサイズを見積もるには、新しいブート環境の作成を開始し、サイズが計算されていることを確認します。ここでプロセスを終了できます。

新しいブート環境上のディスクをブートデバイスとして使用する必要があります。システムの中には、ブートデバイスとして機能するディスクを限定するものがあります。ブート制限が適用されるかどうかを判断するには、システムのドキュメントを参照してください。

新しいブート環境を作成する前に、ディスクの準備が必要になることもあります。ディスクが正しくフォーマットされていることを次のように確認します。

- スライスがファイルシステムをコピーできるだけの十分な大きさであることを確認します。
- ブート環境間でコピーするのではなく、共有するディレクトリが入っているファイルシステムを確認します。ディレクトリを共有する場合、そのディレクトリを固有のスライスに配置して新しいブート環境を作成する必要があります。こうすることにより、ディレクトリは、将来のブート環境と共有可能なファイルシステムになります。独立したファイルシステムを作成して共有する方法についての詳細は、[54 ページの「共有可能なファイルシステムのスライスを選択するための指針」](#)を参照してください。

RAID-1 ボリューム(ミラー)を作成する場合の Live Upgrade の要件

Live Upgrade は Solaris Volume Manager テクノロジを使用して、RAID-1 ボリューム(ミラー)であるファイルシステムを持つブート環境のコピーを作成します。Live Upgrade では、Solaris Volume Manager のすべての機能が実装されるわけではありませんが、Solaris Volume Manager の次のコンポーネントが必要になります。

表 3-3 Live Upgrade と RAID-1 ボリュームに必要なコンポーネント

要件	説明	参照先
状態データベースを1つ以上、状態データベースの複製を3つ以上作成する必要があります。	状態データベースでは、Solaris ボリュームマネージャ構成の状態に関する情報がディスクに保存されます。状態データベースは、複製された複数のデータベースコピーの集まりです。各コピーは「状態データベースの複製」と呼ばれます。状態データベースのコピーを作成することで、単一点障害によるデータ損失を防ぐことができます。	状態データベースの作成については、 『Solaris Volume Manager 管理ガイド』の第6章「状態データベース (概要)」 を参照してください。

表 3-3 Live Upgrade と RAID-1 ボリュームに必要なコンポーネント (続き)

要件	説明	参照先
Live Upgrade では、ルート (/) ファイルシステムに単一スライスの連結を持つ RAID-1 ボリューム (ミラー) だけがサポートされます。	連結は RAID-0 ボリュームです。複数のスライスが連結された方式では、利用可能な最初のスライスがいっぱいになるまでそのスライスにデータが書き込まれます。そのスライスがいっぱいになると次のスライスに連結してデータが書き込まれます。RAID-1 ボリュームに含まれている場合を除き、連結にはデータの冗長性はありません。 RAID-1 ボリュームは、最大 3 つの連結から構成されます。	ミラー化されたファイルシステムの作成のガイドラインについては、 51 ページの「ミラー化されたファイルシステムのスライスを選択するための指針」 を参照してください。

パッケージまたはパッチによるシステムのアップグレード

Live Upgrade を使ってパッチやパッケージをシステムに追加できます。Live Upgrade を使用すると、システムのダウンタイムはリブートの時間だけです。luupgrade コマンドを使って、パッチやパッケージを新しいブート環境に追加できます。luupgrade コマンドを使用する場合は、フラッシュアーカイブを使用してパッチやパッケージをインストールすることもできます。



注意 - Live Upgrade でアップグレードしたりパッケージまたはパッチの追加や削除を行ったりするには、パッケージまたはパッチが最新の SVR4 パッケージガイドラインに準拠している必要があります。Oracle のパッケージはこのガイドラインに準拠していますが、サードパーティーベンダーのパッケージがこれに準拠していることを Oracle は保証できません。非準拠のパッケージを追加しようとすると、アップグレード時にパッケージ追加ソフトウェアの障害が発生するか、アクティブブート環境が改変されてしまう可能性があります。

パッケージの要件の詳細については、[付録 C 「その他の SVR4 パッケージ要件 \(リファレンス\)」](#)を参照してください。

インストールの種類	説明	参照先
ブート環境へのパッチの追加	新しいブート環境を作成してから、-t オプションを指定して luupgrade コマンドを実行します。	98 ページの「ブート環境のネットワークインストールイメージに対するパッチの追加またはパッチの削除」
ブート環境へのパッケージの追加	-p オプションを指定して luupgrade コマンドを実行します。	97 ページの「ブート環境のネットワークインストールイメージに対するパッケージの追加またはパッケージの削除」

インストールの種類	説明	参照先
Live Upgrade を使用したフラッシュアーカイブのインストール	アーカイブには、新しいパッケージやパッチがすでに追加されているブート環境の完全なコピーが格納されています。このコピーを複数のシステムにインストールできます。	<ul style="list-style-type: none"> ■ フラッシュアーカイブを作成する方法の詳細については、『Oracle Solaris 10 1/13 インストールガイド: フラッシュアーカイブ (作成とインストール)』の第3章「フラッシュアーカイブの作成(タスク)」を参照してください。 ■ Live Upgrade を使用したフラッシュアーカイブのインストールについては、108 ページの「ブート環境へのフラッシュアーカイブのインストール」を参照してください。

アップグレードおよびパッチ適用に関する制限事項については、『[Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画](#)』の「アップグレードおよびパッチ適用に関する制限事項」を参照してください。

lucreate コマンドを使用したファイルシステムの作成のための指針

lucreate コマンドに `-m` オプションを指定することにより、新しいブート環境に作成する個別のファイルシステムとファイルシステムの数指定できます。作成するファイルシステムの数だけ、このオプションを繰り返し指定する必要があります。`-m` オプションを使ってファイルシステムを作成する場合、次の項目に留意してください。

- `-m` オプションを1個指定して、新しいブート環境のルート(/) ファイルシステムを作成する必要があります。`-m` オプションを指定しないで `lucreate` を実行すると、「Configuration」メニューが表示されます。「Configuration」メニューでは、新しいマウントポイントにファイルを変更して新しいブート環境をカスタマイズできます。
- 現在のブート環境に存在し、`-m` オプションで指定されないすべてのクリティカルファイルシステムは、作成される上位2番目のファイルシステムにマージされます。
- 新しいブート環境には、`-m` オプションで指定されたファイルシステムだけが作成されます。現在のブート環境に複数のファイルシステムがあり、新しいブート環境にも同じ数のファイルシステムを作成する場合は、ファイルシステムごとに1個ずつ `-m` オプションを指定します。

たとえば、`-m` オプションを1回だけ使用した場合、すべてのファイルシステムが指定の場所に格納されます。元のブート環境のすべてのファイルシステムが、`-m` で指定されたファイルシステムにマージされます。`-m` オプションを2回使用する

と、ファイルシステムが2つ作成されます。ルート (/) ファイルシステム、/opt ファイルシステム、/var ファイルシステムがある場合、これらを新しいブート環境に作成するには、それぞれに -m オプションを1個ずつ指定します。

- マウントポイントが重複しないようにしてください。たとえば、ルート (/) ファイルシステムを2つ作成することはできません。

ファイルシステムのスライスを選択するための指針

ブート環境のファイルシステムを作成する場合、ルールは Oracle Solaris OS のファイルシステムを作成する場合と同じルールです。Live Upgrade では、クリティカルファイルシステムに無効な構成を作成してしまうことを回避できません。たとえば、lucreate コマンドを入力して、ルート (/) と /kernel に別々のファイルシステムを作成することができますが、このようにルート (/) ファイルシステムを分割するのは誤りです。

ディスクスライスを作成するときは、スライスがオーバーラップしないように注意してください。スライスのオーバーラップがあると、新しいブート環境を作成したつもりでも、アクティブにした後ブートすることができません。こうしたオーバーラップは、ファイルシステムの破損の原因となります。

Live Upgrade が正しく機能するには、アクティブブート環境の vfstab ファイルの内容が有効で、ルート (/) ファイルシステムのエントリが少なくとも1つは含まれている必要があります。

ルート (/) ファイルシステムのスライスを選択するための指針

非アクティブブート環境を作成する場合は、ルート (/) ファイルシステムがコピーされるスライスを特定する必要があります。ルート (/) ファイルシステムのスライスを選択する場合は、次の項目に留意してください。

- システムをブートできるスライスである。
- 推奨されている最小サイズ以上である。
- アクティブなルート (/) ファイルシステムとは異なる物理ディスクでも同じディスクでもかまわない。
- VxVM (Veritas Volume Manager) のボリュームにすることができる。現在のシステム上に VxVM ボリュームが構成されている場合は、lucreate コマンドを使用して新しいブート環境を作成できます。新しいブート環境にデータをコピーすると、Veritas ファイルシステム構成が失われ、新しいブート環境に UFS ファイルシステムが作成されます。

ミラー化されたファイルシステムのスライスを選択するための指針

新しく作成するブート環境には、物理ディスクスライス、Solaris ボリュームマネージャーのボリューム、および Veritas Volume Manager のボリュームを自由に組み合わせて使用できます。新しいブート環境にコピーされるクリティカルファイルシステムには、次のような種類があります。

- 物理スライス。
- RAID-1 ボリューム (ミラー) に含まれる単一スライスの連結。ルート (/) ファイルシステムが置かれているスライスは、RAID-1 ボリュームでもかまいません。
- RAID-0 ボリュームに含まれる単一スライスの連結。ルート (/) ファイルシステムが置かれているスライスは、RAID-0 ボリュームでもかまいません。

新しいブート環境を作成する際、`lucreate - m` コマンドは、次の 3 種類のデバイスを認識します。

- 物理スライス (/dev/dsk/cwtxdysz)
- Solaris ボリュームマネージャーのボリューム (/dev/md/dsk/dnum)
- Veritas Volume Manager のボリューム (/dev/vx/dsk/volume_name)。現在のシステム上に VxVM ボリュームが構成されている場合は、`lucreate` コマンドを使用して新しいブート環境を作成できます。新しいブート環境にデータをコピーすると、Veritas ファイルシステム構成が失われ、新しいブート環境に UFS ファイルシステムが作成されます。

注 - Veritas VxVM のアップグレードで問題が生じる場合は、[242 ページの「VERITAS VxVm の実行中にアップグレードする方法」](#)を参照してください。

RAID-1 ボリューム (ミラー) ファイルシステムを作成するための一般的な指針

次のガイドラインを使用して、RAID-1 ボリュームが使用中または再同期中でないかどうか、あるいは Live Upgrade ブート環境が使用しているファイルシステムがボリュームに含まれていないかどうかを確認してください。

ボリュームの命名のガイドラインについては、『[Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画](#)』の「RAID ボリューム名の要件とガイドライン」を参照してください。

ボリュームのステータスの確認

ミラーやサブミラーが保守を必要としている場合や使用中である場合、コンポーネントを切り離すことはできません。新しいブート環境を作成して `detach` キーワード

を使用する前に、`metastat` コマンドを実行してください。`metastat` コマンドは、ミラーが再同期の処理中かどうか、または使用中かどうかを確認します。詳細は、[metastat\(1M\)](#) のマニュアルページを参照してください。

ボリュームの切り離しとミラーの再同期

`detach` キーワードを使ってサブミラーを切り離す場合、`lucreate` コマンドは、デバイスが再同期の処理中かどうかを確認します。デバイスが再同期中である場合、サブミラーを切り離すことはできず、エラーメッセージが表示されます。

再同期処理とは、次のような問題のあとで、あるサブミラーから別のサブミラーにデータをコピーする処理のことです。

- サブミラーの障害
- システムのクラッシュ
- オフラインであったサブミラーがオンラインに復帰
- 新しいサブミラーの追加

再同期処理の詳細は、『[Solaris Volume Manager 管理ガイド](#)』の「[RAID-1 ボリューム \(ミラー\) の再同期](#)」を参照してください。

Solaris ボリュームマネージャーのコマンドの使用

非アクティブなブート環境のボリュームを操作するには、Solaris Volume Manager コマンドではなく `lucreate` コマンドを使用します。Solaris ボリュームマネージャーソフトウェアにはブート環境に関する考慮はありませんが、`lucreate` コマンドでは、ブート環境を誤って破棄しないように確認が行われます。たとえば、`lucreate` では、Solaris ボリュームマネージャーのボリュームの上書きや削除が防止されます。

ただし、Solaris ボリュームマネージャーを使って複雑な連結、ストライプ、ミラーなどを作成した場合、それらのボリュームコンポーネントの操作には Solaris ボリュームマネージャーを使用する必要があります。Live Upgrade では、これらのコンポーネントが認識され、使用がサポートされています。Solaris ボリュームマネージャーのコマンドでボリュームコンポーネントを作成、変更、または破棄する前に、`lustatus` コマンドまたは `lufslist` コマンドを実行してください。これらのコマンドを使用すると、Live Upgrade ブート環境で使用されているファイルシステムがどの Solaris Volume Manager ボリュームに含まれているのかを確認できます。

スワップボリュームのスライスを選択するための指針

ここでは、スワップスライスの構成に関する推奨事項と例を示します。

新しいブート環境のスワップの構成

lucreate コマンドの `-m` オプションを使って、3通りの方法でスワップスライスを構成できます。

- スワップスライスを指定しないと、現在のブート環境のスワップスライスが新しいブート環境用に構成されます。
- スワップスライスを1つ以上指定すると、それらのスワップスライスだけが新しいブート環境で使用されます。この場合指定したスワップスライスは2つのブート環境の間で共有されません。
- スワップスライスを共有すると同時に、新しいスライスを追加することもできます。

3通りのスワップ構成の例を次に示します。現在のブート環境では、ルート (`/`) ファイルシステムが `c0t0d0s0` 上に構成されています。スワップファイルシステムは `c0t0d0s1` 上に構成されています。

- 次の例では、スワップスライスを指定していません。新しいブート環境では、ルート (`/`) ファイルシステムが `c0t1d0s0` 上に置かれます。`c0t0d0s1` 上のスワップが、現在のブート環境と新しいブート環境の間で共有されます。

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

- 次の例では、スワップスライスを指定しています。新しいブート環境では、ルート (`/`) ファイルシステムが `c0t1d0s0` 上に置かれます。新しいスワップファイルシステムが `c0t1d0s1` 上に作成されます。現在のブート環境と新しいブート環境の間でスワップスライスは共有されません。

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

- 次の例では、スワップスライスを1つ追加すると同時に、別のスワップスライスを2つのブート環境で共有しています。新しいブート環境では、ルート (`/`) ファイルシステムが `c0t1d0s0` 上に置かれます。新しいスワップスライスが `c0t1d0s1` 上に作成されます。`c0t0d0s1` 上のスワップスライスが、現在のブート環境と新しいブート環境の間で共有されます。

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

スワップの使用中に起きるブート環境作成の失敗

現在のブート環境以外のブート環境によってスワップスライスが使用されている場合、ブート環境の作成は失敗します。`-s` オプションを使って作成されたブート環境の場合、代替ソースブート環境だけはスワップスライスを使用していなくてもかまいませんが、それ以外のブート環境が使用してはいけません。

共有可能なファイルシステムのスライスを選択するための指針

Live Upgrade は、スライスの内容全体を、指定した新しいブート環境のスライスにコピーします。容量とコピーにかかる時間を節約する場合は、そのスライス上に複数のブート環境で共有できるだけの大きなファイルシステムを用意することもできます。ルート (/) や /var など、OS に欠かせないクリティカルファイルシステムは必ずコピーしてください。/home などの非クリティカルファイルシステムは、複数のブート環境で共有できます。共有可能なファイルシステムは、ユーザーによって定義され、アクティブブート環境と新しいブート環境の両方の個々のスワップスライス上に存在していなければなりません。必要に応じて、複数の方法でディスクを再構成できます。

ディスクの再構成	例	参照先
新しいブート環境を作成する前にディスクスライスを作成し直し、そのスライス上に共有可能なファイルシステムを配置することができます。	たとえば、ルート (/) ファイルシステム、/var、/home がすべて同じスライス上にある場合、ディスクを再構成して /home を固有のスライスに配置できます。デフォルトの設定では、新しいブート環境を作成すると、/home はアクティブブート環境と新しいブート環境で共有されます。	format(1M)
ディレクトリを共有する場合、そのディレクトリを固有のスライスに配置する必要があります。こうすることにより、ディレクトリは、そのほかのブート環境と共有可能なファイルシステムになります。lucreate -m コマンドを実行すると、新しいブート環境が作成され、ディレクトリを固有のスライスに配置することができます。しかし、この新しいファイルシステムはまだ元のブート環境と共有できません。再度 lucreate -m コマンドを実行して、もう 1 つ別のブート環境を作成する必要があります。この 2 つの新しいブート環境では、ディレクトリを共有できます。	たとえば、Solaris 9 リリースから Oracle Solaris 10 8/11 リリースにアップグレードし、両方の OS で /home を共有する場合は、lucreate コマンドを -m オプションで実行します。/home を独立したファイルシステムとして専用のスライス上に持つ Solaris 9 リリースを作成できます。次に、再度 lucreate -m コマンドを実行し、そのブート環境を複製します。この 3 番目のブート環境を Oracle Solaris 10 8/11 リリースにアップグレードします。/home は Solaris 9 リリースと Solaris Oracle Solaris 10 8/11 リリース間で共有されます。	共有可能なファイルシステムおよびクリティカルファイルシステムの概要については、 22 ページの「ファイルシステムのタイプ」 を参照してください。

新しいブート環境の内容のカスタマイズ

新しいブート環境を作成するときに、ディレクトリやファイルの一部を新しいブート環境へのコピーから除外できます。ディレクトリを除外した場合は、そのディレクトリ内にある特定のファイルやサブディレクトリが含まれるようにすることが可能です。これらのサブディレクトリは、新しいブート環境にコピーされません。たとえば、/etc/mail にあるすべてのファイルとディレクトリをコピーから除外

するが、`/etc/mail/staff`にあるすべてのファイルとディレクトリは含まれるように指定できます。たとえば、次のコマンドでは、`staff`サブディレクトリが新しいブート環境にコピーされます。

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



注意-ファイル除外オプションは、注意して使用してください。システムに必要なファイルやディレクトリは削除しないでください。

ブート環境の作成時にディレクトリやファイルをカスタマイズする例については、[88 ページの「ブート環境の作成とコンテンツのカスタマイズ」](#)を参照してください。

ブート環境間でのファイルの同期

新しいブート環境に切り替えてアクティブにする準備が整ったら、ただちに新しいブート環境をアクティブにし、リブートします。新たに作成したブート環境を初めてブートするとき、ブート環境間でファイルの同期がとられます。ここでいう「同期」とは、前にアクティブであったブート環境にあるシステムファイルやディレクトリを、ブートされているブート環境にコピーすることです。変更されているファイルやディレクトリがコピーされます。

`/etc/lu/synclist` ファイルへのファイルの追加

Live Upgrade では、変更されているクリティカルなファイルがチェックされます。これらのファイルの内容が2つのブート環境で同じでない場合、そのファイルはアクティブなブート環境から新しいブート環境にコピーされます。同期は、新しいブート環境の作成後に `/etc/passwd` や `/etc/group` などのクリティカルなファイルが変更された場合のために用意されています。

`/etc/lu/synclist` ファイルには、同期するディレクトリやファイルのリストが記述されています。アクティブなブート環境から新しいブート環境にほかのファイルをコピーする場合もあるでしょう。必要に応じて、ディレクトリやファイルを `/etc/lu/synclist` に追加できます。

`/etc/lu/synclist` のリストにないファイルを追加すると、システムをブートできなくなる場合があります。同期処理では、ファイルのコピーとディレクトリの作成だけが行われます。ファイルやディレクトリの削除は行われません。

次の `/etc/lu/synclist` ファイルの例は、このシステムで同期される標準のディレクトリとファイルを示しています。

/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

次の例は、synclist ファイルに追加することが適切だと思われるディレクトリおよびファイルを示しています。

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

synclist ファイルのエントリは、ファイルまたはディレクトリです。2 番目のフィールドは、ブート環境をアクティブ化するときに行われる更新の方法を示します。ファイルの更新には3通りの方法があります。

- **OVERWRITE** - 新しいブート環境のファイルの内容を、アクティブなブート環境のファイルの内容で上書きします。2 番目のフィールドに動作を指定しない場合は、OVERWRITE がデフォルトの動作となります。エントリがディレクトリである場合は、サブディレクトリもすべてコピーされます。すべてのファイルが上書きされます。新しいブート環境では、ファイルの日付、モード、および所有者は前のブート環境のものと同一になります。
- **APPEND** - 新しいブート環境のファイルの末尾に、アクティブなブート環境のファイルの内容を追加します。この追加によってファイル内のエントリが重複することがあります。ディレクトリには APPEND 動作を指定することはできません。新しいブート環境では、ファイルの日付、モード、および所有者は前のブート環境のものと同一になります。
- **PREPEND** - 新しいブート環境のファイルの先頭に、アクティブなブート環境のファイルの内容を追加します。この追加によってファイル内のエントリが重複することがあります。ディレクトリには PREPEND 動作を指定することはできません。新しいブート環境では、ファイルの日付、モード、および所有者は前のブート環境のものと同一になります。

ブート環境間での強制的な同期

新しく作成したブート環境ではじめてブートするときに、Live Upgrade は新しいブート環境と以前にアクティブだったブート環境の同期をとります。初期のブート

と同期のあとは、リクエストされないかぎり Live Upgrade は同期を実行しません。強制的に同期処理を行うには、`-s` オプションを指定して `luactivate` コマンドを実行します。

複数のバージョンの Oracle Solaris OS を使用していると、強制的に同期を行いたい場合があります。 `email` や `passwd/group` などのファイルに加えた変更を、アクティブにするブート環境に反映させたい場合があります。強制的に同期を実行すると、Live Upgrade は、同期をとるファイルの間に競合がないかチェックします。新しいブート環境がブートされ、競合が検出されると、警告が出されます。この場合、ファイルの同期は行われません。このような競合があっても、アクティブ化は正常に終了する可能性があります。新しいブート環境とアクティブなブート環境の両方で同じファイルに変更を加えると、競合が発生する可能性があります。たとえば、元のブート環境で `/etc/passwd` ファイルに変更を加え、新しいブート環境の `/etc/passwd` ファイルに別の変更を加えた場合、同期処理では同期のためにコピーするファイルを選択できません。



注意-以前のアクティブブート環境で発生した変更気付かない場合や、変更を把握できない場合もあるため、このオプションを使用する際は十分注意してください。たとえば、現在のブート環境で Oracle Solaris 10 8/11 ソフトウェアを実行していて、強制的に同期をとってブート環境を Solaris 9 リリースに戻したとします。この場合、Solaris 9 リリースでファイルが変更されることがあります。ファイルは OS のリリースに依存しているため、Solaris 9 リリースのブートは失敗する可能性があります。Solaris Oracle Solaris 10 8/11 のファイルと Solaris 9 のファイルは互換性があるとは限らないからです。

複数のブート環境のブート

システムに複数の OS がインストールされている場合、SPARC と x86 両方のプラットフォームでそれらのブート環境からブートできます。ブートできるブート環境には、Live Upgrade の非アクティブブート環境が含まれます。

- **Solaris 10 10/08** 以降のリリースでは、SPARC システムで ZFS プール内の ZFS ルートファイルシステムをブートできます。ZFS ルートプールの場合、`boot` コマンドに `-L` オプションを指定して、利用できるブート環境の一覧を表示できます。そこからブート環境を選択し、OBP `boot` コマンドに `-Z` オプションを指定して実行すれば、そのブート環境をブートできます。`-Z` オプションは、ZFS ルートプールの新規ブート環境のブートにも使用される `luactivate` コマンドの代替です。`luactivate` コマンドは、主にブート環境の切り替えに使用します。UFS ファイルシステムでは、プライマリ管理インタフェースとして引き続き、OpenBoot PROM OBP を使用します。指定するブートオプションは、OBP のコマンドを使用して選択します。
- **Solaris 10 1/06** 以降のリリースでは、x86 システムの GRUB ブートメニューに、さまざまなブート環境からブートできるインタフェースが備えられました。Solaris 10 10/08 以降のリリースでは、ブートに利用できる ZFS ブート環境の一覧がこの

メニューに表示されます。デフォルトのブート環境が ZFS ファイルシステムで GRUB メニューが表示されている場合、そのままデフォルトのブート環境をブートすることも、他のブート環境を選択してブートすることもできます。GRUB メニューは、ZFS ルートプールの新規ブート環境のブートにも使用される `luactivate` コマンドの代替です。`luactivate` コマンドは、主にブート環境の切り替えに使用します。

SPARC システムと x86 システムの両方で、各 ZFS ルートプールには、デフォルトのルートファイルシステムとして指定されたデータセットがあります。SPARC の場合は `boot` コマンドの入力、x86 の場合は GRUB メニューからのデフォルトの選択により、このデフォルトのルートファイルシステムがブートされます。

注 - GRUB メニューが明示的に変更され、Live Upgrade で設定した項目以外の項目がデフォルトに指定されている場合、そのデフォルトメニューエントリを選択してもプールのデフォルトのルートファイルシステムがブートされない可能性があります。

ブートおよび GRUB ブートメニューの変更の詳細については、次の項目を参照してください。

タスク	情報
GRUB メニューを使ってブート環境をアクティブにする	119 ページの「x86: GRUB メニューを使ってブート環境をアクティブにする方法」
GRUB メニューを使って元のブート環境に戻す	124 ページの「x86: 新しいブート環境のアクティブ化に成功した場合の GRUB メニューを使ったフォールバック」

タスク	情報
ブートおよびブート動作の変更に関する SPARC と x86 の情報、および詳細な手順	<div>『Oracle Solaris の管理: 基本管理』</div> <ul style="list-style-type: none">■ 『Oracle Solaris の管理: 基本管理』の第 8 章「システムのシャットダウンとブートの概要」■ 『Oracle Solaris の管理: 基本管理』の第 9 章「システムのシャットダウンとブート (概要)」■ 『Oracle Solaris の管理: 基本管理』の第 12 章「Oracle Solaris システムのブート (タスク)」■ 『Oracle Solaris の管理: 基本管理』の第 11 章「Oracle Solaris ブート動作の変更 (タスク)」■ 『Oracle Solaris の管理: 基本管理』の第 13 章「Oracle Solaris ブートアーカイブの管理 (タスク)」
ZFS ブート環境のブートに関する概要および詳細な手順	『Oracle Solaris ZFS 管理ガイド』の「ZFS ルートファイルシステムからのブート」

Live Upgrade のキャラクタユーザーインタフェース

Oracle では、キャラクタユーザーインタフェース (CUI) を表示する `lu` コマンドの使用を推奨していません。CUI の基本的なコマンドシーケンス (`lucreate` コマンド、`luupgrade` コマンド、および `luactivate` コマンドなど) は、簡単に使用できます。これらのコマンドの使用手順については、次の章で説明します。

Live Upgrade を使用したブート環境の作成 (タスク)

この章では、Live Upgrade パッケージとパッチをインストールする方法と、ブート環境を作成する方法について説明します。

注 - この章では、UFS ファイルシステム用の Live Upgrade について説明します。UFS ファイルシステムを ZFS ルートプールに移行する、または ZFS ルートプールを作成およびインストールする手順については、[第 12 章「ZFS ルートプールのブート環境の作成」](#)を参照してください。

この章には次のセクションが含まれています。

- [61 ページの「Live Upgrade のインストールとブート環境の作成 \(タスクマップ\)」](#)
- [62 ページの「Live Upgrade のインストール」](#)
- [66 ページの「新しいブート環境の作成」](#)

Live Upgrade のインストールとブート環境の作成 (タスクマップ)

表 4-1 タスクマップ: Live Upgrade の使用

タスク	説明	参照先
Live Upgrade パッケージをインストールします	OS にパッケージをインストールします	62 ページの「Live Upgrade のインストール」
システムにパッチをインストールします	Live Upgrade には、特定のパッチリビジョンセットが必要です	65 ページの「Live Upgrade に必要なパッチのインストール」

表 4-1 タスクマップ: Live Upgrade の使用 (続き)

タスク	説明	参照先
ブート環境を作成します	非アクティブブート環境に ファイルシステムをコピーして 再構成します	66 ページの「新しいブート環境の作成」

Live Upgrade のインストール

Live Upgrade を実行する前に、インストールメディアから最新の Live Upgrade パッケージをインストールし、My Oracle Support ナレッジドキュメント 1004881.1 – Live Upgrade Software Patch Requirements (以前の 206844) に記載されているパッチをインストールする必要があります。現在の OS に Live Upgrade パッケージをインストールし、古いパッケージを削除する必要があります。Live Upgrade パッケージのリリースは、アップグレード後の OS のリリース番号と一致する必要があります。たとえば、現在使用している OS が Solaris 9 リリースで、Oracle Solaris 10 8/11 リリースにアップグレードする場合、Oracle Solaris 10 8/11 リリースの Live Upgrade パッケージをインストールする必要があります。My Oracle Support ナレッジドキュメント 1004881.1 – Live Upgrade Software Patch Requirements (以前の 206844) に記載されているパッチもインストールする必要があります。最新のパッケージとパッチにより、リリースに最新のバグ修正と新機能がすべて含まれるようになります。新しいブート環境の作成に進む前に、システムに関連するすべてのパッチを必ずインストールしてください。

My Oracle Support にある [My Oracle Support](#) ナレッジドキュメント 1004881.1 Live Upgrade Software Patch Requirements (以前の 206844) では、古いパッケージを削除して新しいパッケージをインストールする方法が説明されており、必要なパッチが記載されています。このセクションでは、ナレッジドキュメント 1004881.1 – Live Upgrade Software Patch Requirements (以前の 206844) に記載されている手順が詳しく説明されています。

パッケージをインストールするには、インストール DVD または CD にある `liveupgrade20` コマンドか、`pkgadd` コマンドを使用します。`liveupgrade20` コマンドを使用する場合は、Java ソフトウェアが必要です。システムに Java ソフトウェアがインストールされていない場合は、`pkgadd` コマンドを使用してパッケージをインストールする必要があります。詳細は、My Oracle Support ナレッジドキュメントを参照してください。

このセクションでは、次の手順について説明します。

- 65 ページの「Live Upgrade に必要なパッチのインストール」
- 63 ページの「`pkgadd` コマンドを使用して Live Upgrade をインストールする方法」
- 63 ページの「Oracle Solaris インストールプログラムを使用して Live Upgrade をインストールする方法」

▼ pkgadd コマンドを使用して Live Upgrade をインストールする方法

- 1 既存の Live Upgrade パッケージを削除します。

3 つの Live Upgrade パッケージ SUNWluu、SUNWlur、および SUNWlucfg は、Live Upgrade を使ってアップグレードまたはパッチを適用するのに必要なソフトウェアを構成します。これらのパッケージには、既存のソフトウェア、新しい機能、およびバグ修正が含まれています。Live Upgrade を使用する前に、既存パッケージを削除しないで新しいパッケージをシステムにインストールすると、ターゲットリリースへのアップグレードまたはパッチ適用は失敗します。SUNWlucfg パッケージは、Solaris 10 8/07 リリースの新機能です。Solaris 10 8/07 より前のリリースから Live Upgrade パッケージを使用する場合は、このパッケージを削除する必要はありません。

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- 2 次の順序でパッケージをインストールします。

```
# pkgadd -d path-to-packages SUNWlucfg SUNWlur SUNWluu
```

path-to-packages ソフトウェアパッケージのあるディレクトリの絶対パスを指定します。

- 3 (省略可能) 指定したパッケージが正常にインストールされていることを確認します。

```
# pkgchk -v SUNWlucfg SUNWlur SUNWluu
```

▼ Oracle Solaris インストールプログラムを使用して Live Upgrade をインストールする方法

インストール DVD または CD にある `liveupgrade20` コマンドを使用する場合、Java ソフトウェアが必要です。システムに Java ソフトウェアがインストールされていない場合は、`pkgadd` コマンドを使用してパッケージをインストールする必要があります。詳細は、[63 ページの「pkgadd コマンドを使用して Live Upgrade をインストールする方法」](#)を参照してください。

注 - この手順では、システムで Solaris Volume Manager を実行していると仮定しています。Solaris Volume Manager を使用したリムーバブルメディアの管理の詳細は、『[Oracle Solaris の管理: デバイスとファイルシステム](#)』を参照してください。

- 1 既存の Live Upgrade パッケージを削除します。

3 つの Live Upgrade パッケージ SUNWluu、SUNWlur、および SUNWlucfg は、Live Upgrade を使ってアップグレードまたはパッチを適用するのに必要なソフトウェアを構成します。これらのパッケージには、既存のソフトウェア、新しい機能、およびバグ修

正が含まれています。Live Upgrade を使用する前に、既存パッケージを削除しないで新しいパッケージをシステムにインストールすると、ターゲットリリースへのアップグレードまたはパッチ適用は失敗します。SUMWLucfg パッケージは、Solaris 10 8/07 リリースの新機能です。Solaris 10 8/07 より前のリリースから Live Upgrade パッケージを使用する場合は、このパッケージを削除する必要はありません。

```
# pkgrm SUNWLucfg SUNWluu SUNWlur
```

2 Oracle Solaris Operating System DVD または Oracle Solaris Software - 2 CD を挿入します。

3 インストーラを実行します。

- Oracle Solaris Operating System DVD を使用している場合は、インストーラのあるディレクトリに移動し、インストーラを実行します。

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

Oracle Solaris インストールプログラムの GUI が表示されます。スクリプトを使用する場合は、`-noconsole` オプションと `-nodisplay` オプションを使用することで、GUI が表示されるのを防ぐことができます。

- Oracle Solaris Software - 2 CD を使用している場合は、インストーラを実行します。

```
% ./installer
```

Oracle Solaris インストールプログラムの GUI が表示されます。

4 「インストール形式の選択」パネルで「カスタム」を選択します。

5 「ロケールの選択」パネルで、インストールする言語を選択します。

6 インストールするソフトウェアを選択します。

- DVD の場合、「コンポーネントの選択」パネルの「次へ」をクリックしてパッケージをインストールします。
- CD の場合、「製品の選択」パネルで Live Upgrade の「デフォルトインストール」を選択し、ほかのソフトウェアの選択を解除する場合はそのソフトウェアをクリックします。

7 Oracle Solaris インストールプログラムの指示に従って、ソフトウェアをインストールします。

必要なパッチをインストールする準備が整いました。

Live Upgrade に必要なパッチのインストール



注意 - Live Upgrade が正しく動作するためには、特定の OS バージョン用の特定のパッチリビジョンのセットがインストールされている必要があります。Live Upgrade をインストールまたは実行する前に、これらのパッチをインストールする必要があります。

ナレッジドキュメント 1004881.1 – Live Upgrade Software Patch Requirements (以前の 206844) に記載されているパッチは、いつでも変更されることがあります。これらのパッチにより、Live Upgrade の欠陥が修正される可能性があると同時に、Live Upgrade が依存するコンポーネントの欠陥も修正される可能性があります。Live Upgrade で問題が発生した場合は、最新の Live Upgrade パッチがインストールされていることを確認してください。

x86 のみ - このパッチのセットがインストールされていない場合、Live Upgrade は失敗し、次のエラーメッセージが表示されることがあります。次のエラーメッセージが表示されなくても、必要なパッチがインストールされていない場合があります。Live Upgrade のインストールを試みる前に、MyOracleSupport のナレッジドキュメントに記載されたすべてのパッチがすでにインストール済みであることを必ず確認してください。

```
ERROR: Cannot find or is not
executable: </sbin/biosdev>.
ERROR: One or more patches required by
Live Upgrade has not been installed.
```

<http://support.oracle.com> を調べて最新の更新済みパッチリストであることを確認してください。My Oracle Support で、ナレッジドキュメント 1004881.1 – Live Upgrade Software Patch Requirements (以前の 206844) を検索してください。

Solaris 8 または Solaris 9 OS を実行している場合は、Live Upgrade インストーラを実行できないことがあります。これらのリリースには、Java 2 Runtime Environment の実行に必要なパッチのセットが含まれていません。Live Upgrade インストーラを実行してパッケージをインストールするには、Java 2 Runtime Environment の推奨パッチクラスタが必要です。Live Upgrade パッケージをインストールするには、pkgadd コマンドを使用します。または、Java 2 Runtime Environment 推奨パッチクラスタをインストールします。パッチクラスタは、<http://support.oracle.com> (My Oracle Support) にあります。

▼ 必要なパッチをインストールする方法

- 1 ローカルディスクにパッチを保存する場合は、`/var/tmp/lupatches`などのディレクトリを作成します。
- 2 **My Oracle Support**のWebサイトから、パッチの一覧を取得します。
- 3 `lupatches`などのパッチディレクトリに変更します。
`# cd /var/tmp/lupatches`
- 4 `patchadd`コマンドを使用してパッチをインストールします。
`# patchadd path-to-patches patch-ID patch-ID`
*patch-ID*はパッチ番号です。複数のパッチ名を指定する場合は、スペースで区切ります。

注-パッチはナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) に指定されている順序で適用する必要があります。

- 5 必要に応じてシステムをリブートします。
いくつかのパッチは、有効にするためにリブートする必要があります。
x86のみ: システムをリブートする必要があります。そうしないと、Live Upgrade は失敗します。
`# init 6`
以上で新しいブート環境を正しく作成するために必要なパッケージとパッチがインストールされました。

新しいブート環境の作成

ブート環境を作成すると、クリティカルファイルシステムをアクティブなブート環境から新しいブート環境にコピーできます。必要に応じてディスクを再編成し、ファイルシステムをカスタマイズして、クリティカルファイルシステムを新しいブート環境にコピーするには、`lucreate`コマンドを使用します。

ファイルシステムは、新しいブート環境にコピーする前にカスタマイズできます。このため、クリティカルファイルシステムディレクトリを親のディレクトリにマージすることも、親ディレクトリから分離することも可能になります。ユーザー定義の(共有可能)ファイルシステムは、デフォルトで複数のブート環境で共有されます。ただし、必要に応じて共有可能なファイルシステムをコピーすることもできます。スワップ(共有可能なボリューム)の分割やマージも可能で

す。クリティカルファイルシステムと共有可能ファイルシステムの概要については、[22 ページの「ファイルシステムのタイプ」](#)を参照してください。

注- この章では、UFS ファイルシステム用の Live Upgrade について説明します。UFS ファイルシステムを ZFS ルートプールに移行する、または ZFS ルートプールを作成およびインストールする手順については、[第 12 章「ZFS ルートプールのブート環境の作成」](#)を参照してください。

lucreate コマンドに `-m` オプションを指定することにより、新しいブート環境に作成するファイルシステムとその数を指定できます。作成するファイルシステムの数だけ、このオプションを繰り返し指定する必要があります。たとえば、`-m` オプションを 1 回だけ使用した場合、すべてのファイルシステムが指定の場所に格納されます。元のブート環境のすべてのファイルシステムが、`-m` で指定されたファイルシステムにマージされます。`-m` オプションを 2 回使用すると、ファイルシステムが 2 つ作成されます。`-m` オプションを使ってファイルシステムを作成する場合、次の項目に留意してください。

- `-m` オプションを 1 個指定して、新しいブート環境のルート (`/`) ファイルシステムを作成する必要があります。`-m` オプションを指定しないで `lucreate` を実行すると、「Configuration」メニューが表示されます。「Configuration」メニューでは、新しいマウントポイントにファイルを変更して新しいブート環境をカスタマイズできます。
- 現在のブート環境に存在し、`-m` オプションで指定されないすべてのクリティカルファイルシステムは、作成される上位 2 番目のファイルシステムにマージされます。
- 新しいブート環境には、`-m` オプションで指定されたファイルシステムだけが作成されます。現在のブート環境に複数のファイルシステムがあり、新しいブート環境でも同じ数のファイルシステムにする場合は、作成するファイルシステムごとに 1 個ずつ `-m` オプションを指定します。たとえば、ルート (`/`) ファイルシステム、`/opt` ファイルシステム、`/var` ファイルシステムがある場合、これらを新しいブート環境に作成するには、それぞれに `-m` オプションを 1 個ずつ指定します。
- マウントポイントが重複しないようにしてください。たとえば、ルート (`/`) ファイルシステムを 2 つ作成することはできません。

▼ ブート環境をはじめて作成する方法

- 次のように入力して新しいブート環境を作成します。

```
# lucreate [-A 'BE-description'] -c BE-name \
  -m mountpoint:device[,metadevice]:fs-options [-m ...] -n BE-name
```

-A 'BE-description'

(オプション) ブート環境名 (BE-name) に関連付けられたブート環境の説明を作成できます。記述の長さ、使用できる文字に制限はありません。

-c BE-name

アクティブブート環境に名前 *BE-name* を割り当てます。このオプションは省略可能で、最初のブート環境を作成する場合だけ使用されます。*lucreate* を初めて実行する場合に *-c* オプションを省略すると、デフォルトの名前が作成されます。

デフォルトの名前は、次の基準に従って選択されます。

- 物理ブートデバイスが判別可能な場合は、その物理ブートデバイスのベース名が現在のブート環境の名前になります。

たとえば、物理ブートデバイスが `/dev/dsk/c0t0d0s0` であれば、現在のブート環境には `c0t0d0s0` という名前が与えられます。

- 物理ブートデバイスが判別不可能な場合は、`uname` コマンドの *-s* オプションと *-r* オプションで取得される名前が組み合わせられます。

たとえば、`uname -s` が返す OS の名前が `SunOS`、`uname -r` が返すリリース名が `5.9` の場合、現在のブート環境には `SunOS5.9` という名前が付けられます。

- どちらのコマンドでも名前を決定できない場合、現在のブート環境には `current` という名前が使われます。

注-最初のブート環境を作成した後は、*-c* オプションを指定しても無視されるか、エラーメッセージが表示されます。

- 現在のブート環境と同じ名前を指定すると、このオプションは無視されます。
- 現在のブート環境と異なる名前を指定すると、作成は失敗し、エラーメッセージが表示されます。次の例は、ブート環境の名前によってエラーメッセージが発生するようすを示しています。

```
# 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 ...]

新しいブート環境のファイルシステム構成を `vfstab` ファイルで指定します。*-m* に引数として指定されるファイルシステムは、同じディスク上のファイルシステムでも、複数のディスク上のファイルシステムでも構いません。このオプションは、作成するファイルシステムの数だけ使用します。

- *mountpoint* には、任意の有効なマウントポイント、またはスワップパーティションを示す *-* (ハイフン) を指定できます。
- *device* フィールドには、次のいずれかを指定できます。

- ディスクデバイスの名前。/dev/dsk/cwtxdysz の形式で表されます
- Solaris Volume Manager のボリューム名 (/dev/md/dsk/dnum の形式)
- Veritas Volume Manager のボリューム名 (/dev/md/vxfs/dsk/dnum の形式)
- キーワード merged。指定されたマウントポイントのファイルシステムがその親とマージされることを示します
- fs-options フィールドには、次のいずれかを指定できます。
 - ufs: UFS ファイルシステムを示します。
 - vxfs: Veritas ファイルシステムを示します。
 - swap: スワップボリュームを示します。スワップマウントポイントはハイフン (-) で表します。
 - 論理デバイス (ミラー) であるファイルシステムについては、いくつかのキーワードを使って、そのファイルシステムに対して実行するアクションを指定できます。論理デバイスの作成、論理デバイスの構成変更、論理デバイスの削除などを行うキーワードがあります。これらのキーワードの説明は、[82 ページの「RAID-1 ボリューム \(ミラー\) を持つブート環境を作成する方法」](#)を参照してください。

-n BE-name

作成するブート環境の名前。BE-name は、システム上で一意である必要があります。

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする (ブート可能な状態にする) ことができます。[第 5 章「Live Upgrade によるアップグレード \(タスク\)」](#)を参照してください。

例 4-1 ブート環境の作成

この例では、アクティブブート環境の名前は first_disk です。-m オプションはファイルシステムのマウントポイントを表します。ルート (/) ファイルシステムと /usr ファイルシステムが作成されます。新しいブート環境の名前は second_disk です。mydescription という記述は、名前 second_disk に対応しています。新しいブート環境 second_disk のスワップは、ソースである first_disk から自動的に共有されます。

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

▼ ブート環境を作成しファイルシステムをマージする方法

注 - `lucreate` コマンドに `-m` オプションを指定することにより、新しいブート環境に作成するファイルシステムとその数を指定できます。作成するファイルシステムの数だけ、このオプションを繰り返し指定する必要があります。たとえば、`-m` オプションを1回だけ使用した場合、すべてのファイルシステムが指定の場所に格納されます。元のブート環境のファイルシステムがすべて1つのファイルシステムにマージされます。`-m` オプションを2回使用すると、ファイルシステムが2つ作成されます。

● 入力:

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

(オプション) ブート環境名 (BE_name) の説明を記述できます。記述の長さ、使用できる文字に制限はありません。

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

新しいブート環境のファイルシステム構成を指定します。`-m` に引数として指定されるファイルシステムは、同じディスク上のファイルシステムでも、複数のディスク上のファイルシステムでも構いません。このオプションは、作成するファイルシステムの数だけ使用します。

- `mountpoint` には、任意の有効なマウントポイント、またはスワップパーティションを示す `-` (ハイフン) を指定できます。
- `device` フィールドには、次のいずれかを指定できます。
 - ディスクデバイスの名前。 `/dev/dsk/cwtxdysz` の形式で表されます
 - Solaris ボリュームマネージャーのメタデバイス名。 `/dev/md/dsk/dnum` の形式で表されます
 - Veritas Volume Manager のボリューム名。 `/dev/vx/dsk/volume_name` の形式で表されます
 - キーワード `merged`。指定されたマウントポイントのファイルシステムがその親とマージされることを示します
- `fs_options` フィールドには、次のいずれかを指定できます。
 - `ufs`: UFS ファイルシステムを示します。
 - `vxfs`: Veritas ファイルシステムを示します。
 - `swap`: スワップボリュームを示します。スワップマウントポイントはハイフン (`-`) で表します。

- 論理デバイス (ミラー) であるファイルシステムについては、いくつかのキーワードを使って、そのファイルシステムに対して実行するアクションを指定できます。論理デバイスの作成、論理デバイスの構成変更、論理デバイスの削除などを行うキーワードがあります。これらのキーワードの説明は、[82 ページの「RAID-1 ボリューム \(ミラー\) を持つブート環境を作成する方法」](#)を参照してください。

`-n BE_name`

作成するブート環境の名前。BE_name は、システム上で一意となるように指定する必要があります。

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする (ブート可能な状態にする) ことができます。[第5章「Live Upgrade によるアップグレード \(タスク\)」](#)を参照してください。

例 4-2 ブート環境を作成しファイルシステムをマージする

この例の場合、現在のブート環境には、ルート (/) ファイルシステム、/usr ファイルシステム、/opt ファイルシステムがあります。/opt ファイルシステムは、親ファイルシステム /usr にマージされます。新しいブート環境の名前は second_disk です。mydescription という記述は、名前 second_disk に対応しています。

```
# 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
```

▼ ブート環境を作成してファイルシステムを分割する方法

注- ブート環境のファイルシステムを作成する場合、ルールは Oracle Solaris OS のファイルシステムを作成する場合のルールと同じです。Live Upgrade では、クリティカルファイルシステムに無効な構成を作成してしまうことを回避できません。たとえば、lucreate コマンドを入力して、ルート (/) と /kernel に別々のファイルシステムを作成できますが、このようにルート (/) ファイルシステムを分割することは無効です。

1つのディレクトリを複数のマウントポイントに分割すると、ファイルシステム間でハードリンクが維持されなくなります。たとえば、/usr/stuff1/file が /usr/stuff2/file にハードリンクされている場合に /usr/stuff1 と /usr/stuff2 を別々のファイルシステムに分割すると、ファイル間のリンクは解除されます。lucreate から警告メッセージが表示され、解除されたハードリンクの代わりとなるシンボリックリンクが作成されます。

● 入力:

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

(オプション) ブート環境名 (BE_name) の説明を記述できます。記述の長さ、使用できる文字に制限はありません。

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

新しいブート環境のファイルシステム構成を指定します。-m に引数として指定されるファイルシステムは、同じディスク上のファイルシステムでも、複数のディスク上のファイルシステムでも構いません。このオプションは、作成するファイルシステムの数だけ使用します。

- *mountpoint* には、任意の有効なマウントポイント、またはスワップパーティションを示す - (ハイフン) を指定できます。
- *device* フィールドには、次のいずれかを指定できます。
 - ディスクデバイスの名前。/dev/dsk/cwtxdysz の形式で表されます
 - Solaris ボリュームマネージャーのメタデバイス名。/dev/md/dsk/dnum の形式で表されます
 - Veritas Volume Manager のボリューム名。/dev/vx/dsk/volume_name の形式で表されます
 - キーワード merged。指定されたマウントポイントのファイルシステムがその親とマージされることを示します
- *fs_options* フィールドには、次のいずれかを指定できます。
 - ufs: UFS ファイルシステムを示します。
 - vxfs: Veritas ファイルシステムを示します。
 - swap: スワップボリュームを示します。スワップマウントポイントはハイフン (-) で表します。
 - 論理デバイス (ミラー) であるファイルシステムについては、いくつかのキーワードを使って、そのファイルシステムに対して実行するアクションを指定できます。論理デバイスの作成、論理デバイスの構成変更、論理デバイスの削除などを行うキーワードがあります。これらのキーワードの説明は、[82 ページの「RAID-1 ボリューム \(ミラー\) を持つブート環境を作成する方法」](#) を参照してください。

```
-n BE_name
```

作成するブート環境の名前。BE_name は、システム上で一意となるように指定する必要があります。

例 4-3 ブート環境を作成しファイルシステムを分割する

この例では、前述のコマンドによってルート (/) ファイルシステムを新しいブート環境内の複数のディスクスライスに分割しています。ここでは、/usr、/var、および /opt をすべてルート (/) に置いている次のソースブート環境を想定してください。

```
/dev/dsk/c0t0d0s0 /
```

新しいブート環境で、次に示すように別々のスライスにマウントすることによって、ファイルシステム /usr、/var、/opt を分割します。

```
/dev/dsk/c0t1d0s0 /
```

```
/dev/dsk/c0t1d0s1 /var
```

```
/dev/dsk/c0t1d0s7 /usr
```

```
/dev/dsk/c0t1d0s5 /opt
```

mydescription という記述は、ブート環境名 second_disk に対応しています。

```
# 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
```

次の手順 新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする(ブート可能な状態にする)ことができます。第5章「[Live Upgrade によるアップグレード\(タスク\)](#)」を参照してください。

▼ **ブート環境を作成しスワップを再構成する方法**

スワップスライスは、デフォルトでは複数のブート環境で共有されます。-m オプションでスワップを指定「しない」場合、現在のブート環境と新しいブート環境が同じスワップスライスを共有します。新しいブート環境のスワップを構成し直す場合は、-m オプションを使用してそのブート環境に対してスワップスライスの追加または削除を行なってください。

注- スワップスライスを分割したりマージしたりするには、現在のブート環境 (-s オプション) を使用した場合はソースブート環境) 以外のブート環境では、スワップスライ스가使用中であってはならないという制限があります。ファイルシステムの種類 (スワップ、ufs など) にかかわらず、スワップスライスはほかのブート環境によって使用されている場合、ブート環境の作成は失敗します。

既存のスワップスライスをを使用してブート環境を作成した後、`vfstab` ファイルを編集することができます。

● 入力:

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

(オプション) ブート環境名 (BE_name) の説明を記述できます。記述の長さ、使用できる文字に制限はありません。

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

新しいブート環境のファイルシステム構成を指定します。-m に引数として指定されるファイルシステムは、同じディスク上のファイルシステムでも、複数のディスク上のファイルシステムでも構いません。このオプションは、作成するファイルシステムの数だけ使用します。

- *mountpoint* には、任意の有効なマウントポイント、またはスワップパーティションを示す - (ハイフン) を指定できます。
- *device* フィールドには、次のいずれかを指定できます。
 - ディスクデバイスの名前。/dev/dsk/cwtxdysz の形式で表されます
 - Solaris ポリユーームマネージャーのメタデバイス名。/dev/md/dsk/dnum の形式で表されます
 - Veritas Volume Manager のポリユーーム名。/dev/vx/dsk/volume_name の形式で表されます
 - キーワード `merged`。指定されたマウントポイントのファイルシステムがその親とマージされることを示します
- *fs_options* フィールドには、次のいずれかを指定できます。
 - `ufs`: UFS ファイルシステムを示します。
 - `vxfs`: Veritas ファイルシステムを示します。
 - `swap`: スワップポリユーームを示します。スワップマウントポイントはハイフン (-) で表します。
 - 論理デバイス (ミラー) であるファイルシステムについては、いくつかのキーワードを使って、そのファイルシステムに対して実行するアクションを指定できます。論理デバイスの作成、論理デバイスの構成変更、論理デ

バイスの削除などを行うキーワードがあります。これらのキーワードの説明は、[82 ページの「RAID-1 ボリューム\(ミラー\)を持つブート環境を作成する方法」](#)を参照してください。

`-n BE_name`

作成するブート環境の名前。`BE_name`は、一意となるように指定する必要があります。

スワップが別のスライスまたはデバイスに移動し、新しいブート環境が作成されます。

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする(ブート可能な状態にする)ことができます。[第5章「Live Upgradeによるアップグレード\(タスク\)」](#)を参照してください。

例 4-4 ブート環境を作成しスワップを再構成する

この例の場合、現在のブート環境には、`/dev/dsk/c0t0d0s0`にルート(`/`)、`/dev/dsk/c0t0d0s1`にスワップがあります。新しいブート環境はルート(`/`)を`/dev/dsk/c0t4d0s0`にコピーし、`/dev/dsk/c0t0d0s1`と`/dev/dsk/c0t4d0s1`の両方をスワップスライスとして使用します。`mydescription`という記述は、ブート環境名`second_disk`に対応しています。

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

これらのスワップ割り当ては、`second_disk`からブートが行われて初めて有効になります。スワップスライスが多数存在する場合は、`-M`オプションを使用してください。[75 ページの「リストを使用してブート環境を作成しスワップを再構成する方法」](#)を参照してください。

▼ リストを使用してブート環境を作成しスワップを再構成する方法

スワップスライスが多数存在する場合は、スワップリストを作成してください。`lucreate`は、新しいブート環境のスワップスライスにこのリストを使用します。

注- スワップスライスを分割したりマージしたりするには、現在のブート環境 (-s オプションを使用した場合はソースブート環境) 以外のブート環境では、スワップスライ스가使用中であってはならないという制限があります。スワップスライスのファイルシステムの種類 (swap、ufs) などにかかわらず、スワップスライ스가ほかのブート環境によって使用されている場合、ブート環境の作成は失敗します。

- 1 新しいブート環境で使用されるスワップスライスのリストを作成します。
このファイルの場所と名前はユーザーが決定できます。この例では、/etc/lu/swapslices ファイルの内容は、デバイスとスライスのリストです。

```
--:/dev/dsk/c0t3d0s2:swap
--:/dev/dsk/c0t3d0s2:swap
--:/dev/dsk/c0t4d0s2:swap
--:/dev/dsk/c0t5d0s2:swap
--:/dev/dsk/clt3d0s2:swap
--:/dev/dsk/clt4d0s2:swap
--:/dev/dsk/clt5d0s2:swap
```

- 2 入力:

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

```
-A 'BE_description'
```

(オプション) ブート環境名 (BE_name) の説明を記述できます。記述の長さ、使用できる文字に制限はありません。

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

新しいブート環境のファイルシステム構成を指定します。-m に引数として指定されるファイルシステムは、同じディスク上のファイルシステムでも、複数のディスク上のファイルシステムでも構いません。このオプションは、作成するファイルシステムの数だけ使用します。

- *mountpoint* には、任意の有効なマウントポイント、またはスワップパーティションを示す - (ハイフン) を指定できます。
- *device* フィールドには、次のいずれかを指定できます。
 - ディスクデバイスの名前。/dev/dsk/cwtxdysz の形式で表されます
 - Solaris ボリュームマネージャーのメタデバイス名。/dev/md/dsk/dnum の形式で表されます
 - Veritas Volume Manager のボリューム名。/dev/vx/dsk/volume_name の形式で表されます
 - キーワード merged。指定されたマウントポイントのファイルシステムがその親とマージされることを示します
- *fs_options* フィールドには、次のいずれかを指定できます。
 - ufs: UFS ファイルシステムを示します。

- `vxfs`: Veritas ファイルシステムを示します。
- `swap`: スワップボリュームを示します。スワップマウントポイントはハイフン(-) で表します。
- 論理デバイス (ミラー) であるファイルシステムについては、いくつかのキーワードを使って、そのファイルシステムに対して実行するアクションを指定できます。論理デバイスの作成、論理デバイスの構成変更、論理デバイスの削除などを行うキーワードがあります。これらのキーワードの説明は、[82 ページの「RAID-1 ボリューム \(ミラー\) を持つブート環境を作成する方法」](#)を参照してください。

`-M slice_list`

ファイル `-slice_list` 中には、`m` オプションのリストが記述されています。これらの引数は、`-m` に指定されている書式で指定してください。ハッシュ記号 (#) で始まるコメント行は無視されます。`-M` オプションは、ブート環境用のファイルシステムが多数存在する場合に便利です。`-m` および `-M` オプションは組み合わせることができます。たとえば、`slice_list` にスワップスライスを記録しておき、`-m` を使用して、ルート (/) スライスと `/usr` スライスを指定できます。

`-m` オプションと `-M` オプションでは、特定のマウントポイントについて複数のスライスを指定できます。これらのスライスを処理する場合、`lucreate` は利用不可能なスライスをスキップして利用できる最初のスライスを選択します。

`-n BE_name`

作成するブート環境の名前。`BE_name` は、一意となるように指定する必要があります。

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする (ブート可能な状態にする) ことができます。[第 5 章「Live Upgrade によるアップグレード \(タスク\)」](#)を参照してください。

例 4-5 リストを使用してブート環境を作成しスワップを再構成する

この例では、新しいブート環境のスワップは、`/etc/lu/swapslices` ファイルに挙げられている一連のスライスです。`mydescription` という記述は、名前 `second_disk` に対応しています。

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

▼ ブート環境を作成し共有可能ファイルシステムをコピーする方法

新しいブート環境に共有可能ファイルシステムをコピーする場合は、`-m` オプションを使用してマウントポイントがコピーされるように指定してください。それ以外の場合、共有可能なファイルシステムはデフォルトで共有され、`vfstab` ファイルに指定された同じマウントポイントを使用します。共有可能ファイルシステムに対する更新は、両方のブート環境に適用されます。

- ブート環境を作成します。

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

(オプション) ブート環境名 (BE_name) の説明を記述できます。記述の長さ、使用できる文字に制限はありません。

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

新しいブート環境のファイルシステム構成を指定します。`-m` に引数として指定されるファイルシステムは、同じディスク上のファイルシステムでも、複数のディスク上のファイルシステムでも構いません。このオプションは、作成するファイルシステムの数だけ使用します。

- `mountpoint` には、任意の有効なマウントポイント、またはスワップパーティションを示す `-` (ハイフン) を指定できます。
- `device` フィールドには、次のいずれかを指定できます。
 - ディスクデバイスの名前。`/dev/dsk/cwtxdysz` の形式で表されます
 - Solaris ボリュームマネージャーのメタデバイス名。`/dev/md/dsk/dnum` の形式で表されます
 - Veritas Volume Manager のボリューム名。`/dev/vx/dsk/volume_name` の形式で表されます
 - キーワード `merged`。指定されたマウントポイントのファイルシステムがその親とマージされることを示します
- `fs_options` フィールドには、次のいずれかを指定できます。
 - `ufs`: UFS ファイルシステムを示します。
 - `vxfs`: Veritas ファイルシステムを示します。
 - `swap`: スワップボリュームを示します。スワップマウントポイントはハイフン (`-`) で表します。
 - 論理デバイス (ミラー) であるファイルシステムについては、いくつかのキーワードを使って、そのファイルシステムに対して実行するアクションを指定できます。論理デバイスの作成、論理デバイスの構成変更、論理デ

バイスの削除などを行うキーワードがあります。これらのキーワードの説明は、82 ページの「RAID-1 ボリューム(ミラー)を持つブート環境を作成する方法」を参照してください。

`-n BE_name`

作成するブート環境の名前。`BE_name` は、一意となるように指定する必要があります。

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする(ブート可能な状態にする)ことができます。第5章「Live Upgrade によるアップグレード(タスク)」を参照してください。

例 4-6 ブート環境を作成し共有可能ファイルシステムをコピーする

この例の場合、現在のブート環境には、ルート(/) ファイルシステムと/home ファイルシステムがあります。新しいブート環境では、ルート(/) ファイルシステムがルート(/) と /usr の2つのファイルシステムに分割されます。/home ファイルシステムは新しいブート環境にコピーされます。mydescription という記述は、ブート環境名 second_disk に対応しています。

```
# 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
```

▼ 別のソースからブート環境を作成する方法

lucreate コマンドは、アクティブブート環境内のファイルシステムに基づいてブート環境を作成します。アクティブブート環境以外のブート環境に基づいてブート環境を作成する場合は、`-s` オプションを指定して lucreate を実行します。

注-新しいブート環境をアクティブにした後、フォールバックを行う必要がある場合は、ソースブート環境ではなく以前にアクティブだったブート環境に戻ります。

- ブート環境を作成します。

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

(オプション) ブート環境名 (BE_name) の説明を記述できます。記述の長さ、使用できる文字に制限はありません。

`-s source_BE_name`

新しいブート環境に対するソースブート環境を指定します。このソースはアクティブブート環境ではありません。

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

新しいブート環境のファイルシステム構成を指定します。-m に引数として指定されるファイルシステムは、同じディスク上のファイルシステムでも、複数のディスク上のファイルシステムでも構いません。このオプションは、作成するファイルシステムの数だけ使用します。

- *mountpoint* には、任意の有効なマウントポイント、またはスワップパーティションを示す - (ハイフン) を指定できます。
- *device* フィールドには、次のいずれかを指定できます。
 - ディスクデバイスの名前。/dev/dsk/cwtxdysz の形式で表されます
 - Solaris ボリュームマネージャーのメタデバイス名。/dev/md/dsk/dnum の形式で表されます
 - Veritas Volume Manager のボリューム名。/dev/vx/dsk/volume_name の形式で表されます
 - キーワード *merged*。指定されたマウントポイントのファイルシステムがその親とマージされることを示します
- *fs_options* フィールドには、次のいずれかを指定できます。
 - *ufs*: UFS ファイルシステムを示します。
 - *vxfs*: Veritas ファイルシステムを示します。
 - *swap*: スワップボリュームを示します。スワップマウントポイントはハイフン (-) で表します。
 - 論理デバイス(ミラー)であるファイルシステムについては、いくつかのキーワードを使って、そのファイルシステムに対して実行するアクションを指定できます。論理デバイスの作成、論理デバイスの構成変更、論理デバイスの削除などを行うキーワードがあります。これらのキーワードの説明は、[82 ページの「RAID-1 ボリューム\(ミラー\)を持つブート環境を作成する方法」](#)を参照してください。

`-n BE_name`

作成するブート環境の名前。*BE_name* は、システム上で一意となるように指定する必要があります。

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする(ブート可能な状態にする)ことができます。[第5章「Live Upgradeによるアップグレード\(タスク\)」](#)を参照してください。

例 4-7 別のソースから単一のブート環境を作成する

この例では、*third_disk* という名前のソースブート環境内のルート (/) ファイルシステムに基づいてブート環境を作成します。*third_disk* はアクティブブート環境ではありません。「*mydescription*」という記述は、*second_disk* という名前の新しいブート環境に対応しています。


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

▼ フラッシュアーカイブ用の空のブート環境を作成する方法

lucreate コマンドは、アクティブブート環境内のファイルシステムに基づいてブート環境を作成します。lucreate コマンドに `-s` オプションを指定して実行すると、空のブート環境を短時間で作成できます。スライスは、指定のファイルシステム用に予約されていますが、ファイルシステムはコピーされません。このブート環境は、名前が付けられますが、フラッシュアーカイブがインストールされるまでは実際には作成されません。空のブート環境にアーカイブがインストールされると、ファイルシステムは予約されたスライスにインストールされます。

- 空のブート環境を作成します。

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

(オプション) ブート環境名 (BE_name) の説明を記述できます。記述の長さ、使用できる文字に制限はありません。

`-s -`

空のブート環境を作成します。

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

新しいブート環境のファイルシステム構成を指定します。`-m` に引数として指定されるファイルシステムは、同じディスク上のファイルシステムでも、複数のディスク上のファイルシステムでも構いません。このオプションは、作成するファイルシステムの数だけ使用します。

- `mountpoint` には、任意の有効なマウントポイント、またはスワップパーティションを示す `-` (ハイフン) を指定できます。
- `device` フィールドには、次のいずれかを指定できます。
 - ディスクデバイスの名前。/dev/dsk/cwtxdysz の形式で表されます
 - Solaris ボリュームマネージャーのメタデバイス名。/dev/md/dsk/dnum の形式で表されます
 - Veritas Volume Manager のボリューム名。/dev/vx/dsk/volume_name の形式で表されます
 - キーワード `merged`。指定されたマウントポイントのファイルシステムがその親とマージされることを示します
- `fs_options` フィールドには、次のいずれかを指定できます。
 - `ufs`: UFS ファイルシステムを示します。

- **vxfs**: Veritas ファイルシステムを示します。
- **swap**: スワップボリュームを示します。スワップマウントポイントはハイフン(-) で表します。
- 論理デバイス(ミラー)であるファイルシステムについては、いくつかのキーワードを使って、そのファイルシステムに対して実行するアクションを指定できます。論理デバイスの作成、論理デバイスの構成変更、論理デバイスの削除などを行うキーワードがあります。これらのキーワードの説明は、[82 ページの「RAID-1 ボリューム\(ミラー\)を持つブート環境を作成する方法」](#)を参照してください。

-n BE_name

作成するブート環境の名前。BE_name は、システム上で一意となるように指定する必要があります。

例 4-8 フラッシュアーカイブ用の空のブート環境を作成する

この例では、ファイルシステムを一切含まないブート環境を作成します。「mydescription」という記述は、second_disk という名前の新しいブート環境に対応しています。

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

空のブート環境の作成が完了したら、フラッシュアーカイブをインストールし、アクティブ(ブート可能な状態)にすることができます。[第 5 章「Live Upgrade によるアップグレード\(タスク\)」](#)を参照してください。

空のブート環境の作成とデータの取り込みの例については、[172 ページの「空のブート環境を作成してフラッシュアーカイブをインストールする例」](#)を参照してください。

▼ RAID-1 ボリューム(ミラー)を持つブート環境を作成する方法

ブート環境を作成するとき、Live Upgrade は Solaris Volume Manager テクノロジーを使って RAID-1 ボリュームを作成します。ブート環境を作成するとき、Live Upgrade を使って次のタスクを行うことができます。

- 単スライスの連結(サブミラー)を RAID-1 ボリューム(ミラー)から切り離します。必要な場合は、内容を保存して新しいブート環境の内容にすることができます。内容はコピーされないため、新しいブート環境を短時間で作成できます。ミラーから切り離されたサブミラーは、元のミラーの一部ではなくなります。サブミラーに対する読み取りや書き込みがミラーを介して実行されることはなくなります。

- ミラーを含んだブート環境を作成します。
- 新しく作成したミラーに単一スライスの連結を接続します。

始める前に Live Upgrade のミラー化機能を使用するには、状態データベースと状態データベースの複製を作成する必要があります。状態データベースでは、Solaris ボリュームマネージャー構成の状態に関する情報がディスクに保存されます。

- 状態データベースの作成については、『[Solaris Volume Manager 管理ガイド](#)』の第6章「[状態データベース \(概要\)](#)」を参照してください。
 - Solaris Volume Manager の概要と、Live Upgrade で実行できるタスクについては、[27 ページの「RAID-1 ボリュームファイルシステムを持つブート環境の作成」](#)を参照してください。
 - Live Upgrade を使用するときには使用できない Solaris Volume Manager の複雑な構成についての詳細は、『[Solaris Volume Manager 管理ガイド](#)』の第2章「[ストレージ管理の概念](#)」を参照してください。
- 次のように入力して新しいブート環境を作成します。

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

-A 'BE_description'

(オプション) ブート環境名 (BE_name) の説明を記述できます。記述の長さ、使用できる文字に制限はありません。

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

新しいブート環境のファイルシステム構成を `vfstab` で指定します。-m に引数として指定されるファイルシステムは、同じディスク上のファイルシステムでも、複数のディスク上のファイルシステムでも構いません。このオプションは、作成するファイルシステムの数だけ使用します。

- `mountpoint` には、任意の有効なマウントポイント、またはスワップパーティションを示す - (ハイフン) を指定できます。
- `device` フィールドには、次のいずれかを指定できます。
 - ディスクデバイスの名前。/dev/dsk/cwtxdysz の形式で表されます
 - Solaris Volume Manager のボリューム名 (/dev/md/dsk/dnum の形式)
 - Veritas Volume Manager のボリューム名 (/dev/md/vxfs/dsk/dnum の形式)
 - キーワード `merged`。指定されたマウントポイントのファイルシステムがその親とマージされることを示します
- `fs_options` には、次のいずれかの種類のファイルシステムとキーワードを指定できます。
 - `ufs`: UFS ファイルシステムを示します。
 - `vxfs`: Veritas ファイルシステムを示します。

- **swap**: スワップボリュームを示します。スワップマウントポイントはハイフン (-) で表します。
- 論理デバイス (ミラー) であるファイルシステムについては、いくつかのキーワードを使って、そのファイルシステムに対して実行するアクションを指定できます。論理デバイスの作成、論理デバイスの構成変更、論理デバイスの削除などを行うキーワードがあります。
 - **mirror** を指定すると、指定したデバイスに RAID-1 ボリューム (ミラー) を作成できます。その後の **-m** オプションで **attach** を指定して、少なくとも1つの連結を新しいミラーに接続する必要があります。指定するデバイスには、正しく名前が付けられている必要があります。たとえば、論理デバイスの名前 **/dev/md/dsk/d10** をミラー名として使用できます。デバイスの命名の詳細は、『[Solaris Volume Manager 管理ガイド](#)』の「[Solaris Volume Manager コンポーネントの概要](#)」を参照してください。
 - **detach** を指定すると、指定したマウントポイントに関連付けられているボリュームから連結を切り離すことができます。ボリュームを指定する必要はありません。
 - **attach** を指定すると、指定したマウントポイントに関連付けられているミラーに連結を接続できます。指定した物理ディスクスライスは、単一デバイスの連結になり、ミラーに接続されます。ディスクに接続する連結を指定するには、デバイス名の後ろにコンマと連結の名前を付加します。コンマと連結の名前を省略して **lucreate** を実行すると、空いているボリュームが連結用に選択されます。

lucreate で作成できるのは、単一の物理スライスから成る連結だけです。このコマンドでは、1つのミラーに3つまで連結を接続できます。

- **preserve** を指定すると、既存のファイルシステムとその内容を保存できます。このキーワードを使うと、ソースブート環境の内容をコピーする処理を省略できます。内容を保存することで、新しいブート環境を短時間で作成できます。特定のマウントポイントについて、**preserve** で指定できるのは1つの物理デバイスだけです。**preserve** を指定して **lucreate** コマンドを実行すると、指定したファイルシステムに対してデバイスの内容が適切かどうかを検査されます。この検査は限定的であり、適合性を保証することはできません。

preserve キーワードは、物理スライスと Solaris ボリュームマネージャのボリュームの両方に使用できます。

- UFS ファイルシステムが物理スライスに置かれている場合に **preserve** キーワードを使用すると、UFS ファイルシステムの内容がそのスライスに保存されます。次の **-m** オプションの例では、**preserve** キーワードを使って、物理デバイス **c0t0d0s0** の内容を、ルート (**/**) ファイルシステムのマウントポイント用のファイルシステムとして保存します。

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

- UFS ファイルシステムがボリュームに置かれている場合に `preserve` キーワードを使用すると、UFS ファイルシステムの内容がそのボリュームに保存されます。

次の `-m` オプションの例では、`preserve` キーワードを使って、RAID-1 ボリューム (ミラー) `d10` の内容を、ルート (`/`) ファイルシステムのマウントポイント用のファイルシステムとして保存します。

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

次の `-m` オプションの例では、RAID-1 ボリューム (ミラー) `d10` がルート (`/`) ファイルシステムのマウントポイント用のファイルシステムとして構成されます。単一スライスの連結 `d20` が現在のミラーから切り離されます。`d20` がミラー `d10` に接続されます。ルート (`/`) ファイルシステムは、サブミラー `d20` に保持されます。

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

```
-n BE_name
```

作成するブート環境の名前。`BE_name` は、システム上で一意となるように指定する必要があります。

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする (ブート可能な状態にする) ことができます。第 5 章「[Live Upgrade によるアップグレード \(タスク\)](#)」を参照してください。

例 4-9 ミラーを持つブート環境の作成とデバイスの指定

この例では、ファイルシステムのマウントポイントを `-m` オプションで指定します。

- 「`mydescription`」という記述は、`another_disk` に対応しています。
- `lucreate` コマンドにより、ルート (`/`) マウントポイントの UFS ファイルシステムが構成されます。`d10` というミラーが作成されます。このミラー `d10` に、現在のブート環境のルート (`/`) ファイルシステムがコピーされます。ミラー `d10` にあるデータはすべて上書きされます。
- 2つのスライス `c0t0d0s0` および `c0t1d0s0` は、サブミラー `d1` および `d2` として指定されています。これら2つのサブミラーは、ミラー `d10` に追加されます。
- 新しいブート環境には `another_disk` という名前が付けられます。

```
# 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 -n another_disk
```

例 4-10 ミラーを持つブート環境の作成とサブミラー名の省略

この例では、ファイルシステムのマウントポイントを `-m` オプションで指定します。

- 「mydescription」という記述は、another_disk に対応しています。
- `lucreate` コマンドにより、ルート (`/`) マウントポイントの UFS ファイルシステムが構成されます。d10 というミラーが作成されます。このミラー d10 に、現在のブート環境のルート (`/`) ファイルシステムがコピーされます。ミラー d10 にあるデータはすべて上書きされます。
- 2つのスライス `c0t0d0s0` および `c0t1d0s0` は、サブミラーとして指定されています。サブミラーを指定せずに `lucreate` コマンドを実行すると、利用可能なボリューム名の一覧から名前が選択されます。これら 2つのサブミラーは、ミラー d10 に接続されます。
- 新しいブート環境には another_disk という名前が付けられます。

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

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする(ブート可能な状態にする)ことができます。第 5 章「[Live Upgrade によるアップグレード\(タスク\)](#)」を参照してください。

例 4-11 ブート環境の作成とサブミラーの切り離し

この例では、ファイルシステムのマウントポイントを `-m` オプションで指定します。

- 「mydescription」という記述は、another_disk に対応しています。
- `lucreate` コマンドにより、ルート (`/`) マウントポイントの UFS ファイルシステムが構成されます。d10 というミラーが作成されます。
- スライス `c0t0d0s0` が現在のミラーから切り離されます。このスライスはサブミラー d1 として指定され、ミラー d10 に追加されます。このサブミラーの内容であるルート (`/`) ファイルシステムは保存され、コピー処理は発生しません。スライス `c0t1d0s0` はサブミラー d2 として指定され、ミラー d10 に追加されます。
- 新しいブート環境には another_disk という名前が付けられます。

```
# 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
```

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする(ブート可能な状態にする)ことができます。第 5 章「[Live Upgrade によるアップグレード\(タスク\)](#)」を参照してください。

例 4-12 ブート環境の作成、サブミラーの切り離しと内容の保存

この例では、ファイルシステムのマウントポイントを `-m` オプションで指定します。

- 「mydescription」という記述は、another_disk に対応しています。
- `lucreate` コマンドにより、ルート (`/`) マウントポイントの UFS ファイルシステムが構成されます。d20 というミラーが作成されます。
- スライス `c0t0d0s0` が現在のミラーから切り離され、ミラー d20 に追加されます。サブミラーの名前は指定されていません。このサブミラーの内容であるルート (`/`) ファイルシステムは保存され、コピー処理は発生しません。
- 新しいブート環境には another_disk という名前が付けられます。

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

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする(ブート可能な状態にする)ことができます。第 5 章「[Live Upgrade によるアップグレード\(タスク\)](#)」を参照してください。

例 4-13 2つのミラーを持つブート環境の作成

この例では、ファイルシステムのマウントポイントを `-m` オプションで指定します。

- 「mydescription」という記述は、another_disk に対応しています。
- `lucreate` コマンドにより、ルート (`/`) マウントポイントの UFS ファイルシステムが構成されます。d10 というミラーが作成されます。このミラー d10 に、現在のブート環境のルート (`/`) ファイルシステムがコピーされます。ミラー d10 にあるデータはすべて上書きされます。
- 2つのスライス `c0t0d0s0` および `c0t1d0s0` は、サブミラー d1 および d2 として指定されています。これら2つのサブミラーは、ミラー d10 に追加されます。
- `lucreate` コマンドにより、`/opt` マウントポイントの UFS ファイルシステムが構成されます。d11 というミラーが作成されます。このミラー d11 に、現在のブート環境の `/opt` ファイルシステムがコピーされます。ミラー d11 にあるデータはすべて上書きされます。
- 2つのスライス `c2t0d0s1` および `c3t1d0s1` は、サブミラー d3 および d4 として指定されています。これら2つのサブミラーは、ミラー d11 に追加されます。
- 新しいブート環境には another_disk という名前が付けられます。

```
# 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
```

新しいブート環境の作成が終わると、この環境をアップグレードしてアクティブにする(ブート可能な状態にする)ことができます。第5章「[Live Upgradeによるアップグレード\(タスク\)](#)」を参照してください。

ブート環境の作成とコンテンツのカスタマイズ

新しいブート環境のファイルシステムの内容は、追加用オプションおよび除外用オプションを使用して変更できます。ディレクトリやファイルは新しいブート環境にコピーされません。

追加用オプションおよび除外用オプションは、`lucreate` コマンド構文では次のようになります。

```
# 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
```

-x *exclude-dir*

ファイルやディレクトリを除外して、新しいブート環境にコピーされないようにします。このオプションのインスタンスを複数使用して、複数のファイルまたはディレクトリを除外できます。

exclude-dir はディレクトリまたはファイルの名前です。

-y *include-dir*

指定されたディレクトリやファイルを新しいブート環境にコピーします。ディレクトリは除外したが、個々のサブディレクトリまたはファイルを含める場合、このオプションを使用します。

include-dir には、含めるサブディレクトリやファイルの名前を指定します。

-Y *list-filename*

リスト内のディレクトリやファイルを新しいブート環境にコピーします。ディレクトリは除外したが、個々のサブディレクトリまたはファイルを含める場合、このオプションを使用します。

- *list-filename* は、リストを含むファイルのフルパスです。
- *list-filename* ファイルでは、1 ファイルを1 行で表す必要があります。
- 行でディレクトリを指定すると、そのディレクトリの下にあるすべてのサブディレクトリとファイルが含まれます。行でファイルを指定すると、そのファイルだけが含まれます。

-f *list-filename*

リスト内のファイルやディレクトリを除外して、新しいブート環境にコピーされないようにします。

- *list-filename* は、リストを含むファイルのフルパスです。
- *list-filename* ファイルでは、1 ファイルを1行で表す必要があります。

-z *list-filename*

リスト内のディレクトリやファイルを新しいブート環境にコピーします。リスト内の各ファイルまたはディレクトリには、プラス「+」またはマイナス「-」記号を付けます。プラスはファイルやディレクトリを含めることを、マイナスはファイルやディレクトリを除外することを示します。

- *list-filename* は、リストを含むファイルのフルパスです。
- *list-filename* ファイルでは、1 ファイルを1行で表す必要があります。プラスまたはマイナスとファイル名との間には1つの半角スペースが必要です。
- 行でディレクトリとプラス(+)を指定すると、そのディレクトリの下にあるすべてのサブディレクトリとファイルがコピーされます。行でファイルとプラス(+)を指定すると、そのファイルだけがコピーされます。

-I

システムファイルの整合性チェックをオーバーライドします。このオプションは慎重に使用してください。

重要なシステムファイルをブート環境から除外してしまうことを防ぐために、lucreate は整合性チェックを実行します。このチェックにより、システムパッケージデータベースに登録されたすべてのファイルが検査され、そのいずれかが除外されると、ブート環境の作成が停止します。このオプションを指定すると、この整合性チェックをオーバーライドします。このオプションを指定すると、より短時間でブート環境を作成できますが、問題を検出できなくなる可能性があります。

Live Upgrade によるアップグレード (タスク)

この章では、Live Upgrade を使用して非アクティブブート環境のアップグレードとアクティブ化を行う方法について説明します。

注- この章では、UFS ファイルシステム用の Live Upgrade について説明します。luupgrade コマンドと luactivate コマンドの使用法は、ZFS ブート環境でも同じです。UFS ファイルシステムを ZFS ルートプールに移行する、または ZFS ルートプールを作成およびインストールする手順については、[第 12 章「ZFS ルートプールのブート環境の作成」](#)を参照してください。

この章には次のセクションが含まれています。

- [91 ページの「ブート環境のアップグレード \(タスクマップ\)」](#)
- [92 ページの「ブート環境のアップグレード」](#)
- [108 ページの「ブート環境へのフラッシュアーカイブのインストール」](#)
- [113 ページの「ブート環境のアクティブ化」](#)

ブート環境のアップグレード (タスクマップ)

表 5-1 タスクマップ: Live Upgrade を使ったアップグレード

タスク	説明	参照先
ブート環境をアップグレードするか、フラッシュアーカイブをインストールします。	<ul style="list-style-type: none">■ OS イメージを使用して非アクティブブート環境をアップグレードします。■ 非アクティブブート環境にフラッシュアーカイブをインストールします。	<ul style="list-style-type: none">■ 92 ページの「ブート環境のアップグレード」■ 108 ページの「ブート環境へのフラッシュアーカイブのインストール」

表 5-1 タスクマップ: Live Upgrade を使ったアップグレード (続き)

タスク	説明	参照先
非アクティブブート環境をアクティブにします。	変更を有効にし、非アクティブブート環境をアクティブに切り替えます。	113 ページの「ブート環境のアクティブ化」
(オプション) アクティブ化で問題が発生した場合に戻します。	問題が発生する場合は元のブート環境をアクティブに戻します。	第 6 章「障害回復: 元のブート環境へのフォールバック (タスク)」

ブート環境のアップグレード

ブート環境をアップグレードするには、`luupgrade` コマンドを使用します。このセクションでは、次の場所に置かれているファイルを使用して非アクティブブート環境をアップグレードする手順について説明します。

- NFS サーバー
- ローカルファイル
- ローカルテープ
- ローカルデバイス (DVD または CD)

アップグレードのガイドライン

最新の OS でブート環境をアップグレードすると、アクティブブート環境には影響を及ぼしません。新しいファイルは非アクティブブート環境のクリティカルファイルシステムとマージされますが、共有可能ファイルシステムは変更されません。

RAID-1 ボリュームがインストールされている場合や非大域ゾーンがインストールされている場合もアップグレードが可能です。または、フラッシュアーカイブをインストールすることもできます。

- 物理ディスクスライス、Solaris ボリュームマネージャーのボリューム、および Veritas Volume Manager のボリュームがどのような組み合わせで含まれているブート環境もアップグレードできます。ルート (/) ファイルシステム用のスライスは、RAID-1 ボリューム (ミラー) に含まれている単一スライスの連結でなければいけません。ミラー化されたファイルシステムを持つブート環境の作成については、[82 ページの「RAID-1 ボリューム \(ミラー\) を持つブート環境を作成する方法」](#)を参照してください。

注-現在のシステム上に VxVM ポリウムが構成されている場合は、lucreate コマンドを使用して新しいブート環境を作成できます。新しいブート環境にデータをコピーすると、Veritas ファイルシステム構成が失われ、新しいブート環境に UFS ファイルシステムが作成されます。

- 任意のインストールプログラムで非大域ゾーンがインストールされているシステムもアップグレードできます。非大域ゾーンが含まれるアップグレードの手順については、[第8章「非大域ゾーンがインストールされているシステムにおける Oracle Solaris OS のアップグレード」](#)を参照してください。
- アップグレードを行う代わりに、フラッシュアーカイブを作成して、非アクティブブート環境にアーカイブをインストールできます。それらの新しいファイルは非アクティブブート環境のクリティカルファイルシステムを上書きしますが、共有可能ファイルシステムは変更されません。[108 ページの「ブート環境へのフラッシュアーカイブのインストール」](#)を参照してください。

パッケージまたはパッチによるシステムのアップグレード

Live Upgrade を使ってパッチやパッケージをシステムに追加できます。Live Upgrade では、現在動作しているシステムのコピーが作成されます。この新しいブート環境は、アップグレードすることも、パッケージやパッチを追加することもできます。Live Upgrade を使用すると、システムのダウンタイムはリブートの時間だけですみます。luupgrade コマンドを使って、パッチやパッケージを新しいブート環境に追加できます。



注意-Live Upgrade でパッケージまたはパッチの追加や削除を行うには、パッケージまたはパッチが最新の SVR4 パッケージガイドラインに準拠している必要があります。Oracle のパッケージはこのガイドラインに準拠していますが、サードパーティーベンダーのパッケージがこれに準拠していることを Oracle は保証できません。非準拠のパッケージを追加しようとすると、パッケージ追加ソフトウェアの障害が発生するか、アップグレード中にアクティブブート環境が改変されてしまう可能性があります。

パッケージの要件の詳細については、[付録 C 「その他の SVR4 パッケージ要件 \(リファレンス\)」](#)を参照してください。

表 5-2 パッケージやパッチを使ったブート環境のアップグレード

インストールの種類	説明	参照先
ブート環境へのパッチの追加	新しいブート環境を作成してから、-t オプションを指定して luupgrade コマンドを実行します。	98 ページの「ブート環境のネットワークインストールイメージに対するパッチの追加またはパッチの削除」

表 5-2 パッケージやパッチを使ったブート環境のアップグレード (続き)

インストールの種類	説明	参照先
ブート環境への パッケージの追加	-p オプションを指定して luupgrade コマンドを実行します。	97 ページの「ブート環境のネット ワークインストールイメージに対 するパッケージの追加または パッケージの削除」

▼ ブート環境のネットワークインストールイメージをアップグレードする方法

この手順でアップグレードを行う場合は、DVD またはネットワークインストールイメージを使用する必要があります。インストールに複数の CD が必要な場合は、95 ページの「複数の CD のネットワークインストールイメージをアップグレードする方法」の手順を使用する必要があります。

注 - Oracle Solaris 10 9/10 リリース以降では、DVD のみが提供されます。Oracle Solaris Software CD は提供されません。

- 1 **Live Upgrade** のパッケージ **SUNWlucfg**、**SUNWlur**、および **SUNWluu** をシステムにインストールします。
これらのパッケージは、アップグレード先のリリースのものである必要があります。手順については、63 ページの「**pkgadd コマンドを使用して Live Upgrade をインストールする方法**」を参照してください。
- 2 アップグレードするブート環境とインストールソフトウェアのパスを指定します。

```
# luupgrade -u -n BE-name -s os-image-path
```

-u ブート環境のネットワークインストールイメージをアップグレードします。

-n *BE-name* アップグレード対象のブート環境の名前を指定します。

-s *os-image-path* ネットワークインストールイメージが置かれているディレクトリのパス名を指定します。

例 5-1 DVD メディアからブート環境のネットワークインストールイメージをアップグレードする

この例では、second_disk ブート環境は DVD メディアを使用してアップグレードされます。pkgadd コマンドは、アップブレード先のリリースの Live Upgrade パッケージを追加します。

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

例 5-2 ネットワークインストールイメージからブート環境のネットワークインストールイメージをアップグレードする

この例では、second_disk ブート環境をアップグレードします。pkgadd コマンドは、アップグレード先のリリースの Live Upgrade パッケージを追加します。

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

▼ 複数の CD のネットワークインストールイメージをアップグレードする方法

注 - Oracle Solaris 10 9/10 リリース以降では、DVD のみが提供されます。Oracle Solaris Software CD は提供されません。

ネットワークインストールイメージは複数の CD 上にあるため、このアップグレード手順を使用する必要があります。ほかの CD がある場合は、luupgrade コマンドに -i オプションを指定してそれらの CD をインストールします。

- 1 Live Upgrade のパッケージ SUNWlucfg、SUNWlur、および SUNWluu をシステムにインストールします。

これらのパッケージは、アップグレード先のリリースのものである必要があります。手順については、[63 ページの「pkgadd コマンドを使用して Live Upgrade をインストールする方法」](#)を参照してください。

- 2 アップグレードするブート環境とインストールソフトウェアのパスを指定します。

```
# luupgrade -u -n BE-name -s os-image-path
```

-u ブート環境のネットワークインストールイメージをアップグレードします。

-n BE-name アップグレード対象のブート環境の名前を指定します。

-s os-image-path ネットワークインストールイメージが置かれているディレクトリのパス名を指定します。

- 3 1 枚目の CD のインストーラ処理が完了したら、2 枚目の CD を挿入します。

4 メニューまたはテキストを使用して、2枚目のCD上のインストーラを実行します。

- メニューを使用して2枚目のCD上のインストーラを実行するには:

```
# luupgrade -i -n BE-name -s os-image-path
```

- テキストを使用して2枚目のCD上のインストーラを実行し、ユーザーの操作を不要にするには:

```
# luupgrade -i -n BE-name -s os-image-path -O '-nodisplay -noconsole'
```

-i	追加のCDがインストールされます。ソフトウェアは、指定されたメディア上のインストールプログラムを探し、そのプログラムを実行します。インストーラプログラムは、-sで指定します。
-n BE-name	アップグレード対象のブート環境の名前を指定します。
-s os-image-path	ネットワークインストールイメージが置かれているディレクトリのパス名を指定します。
-O '-nodisplay -noconsole'	(オプション)テキストモードで2枚目のCD上のインストーラを実行します。ユーザーに入力を求めるプロンプトは表示されません。

5 インストールするCDごとに、[手順3](#)および[手順4](#)を繰り返します。

例 5-3 SPARC: 複数のCDを使用してネットワークインストールイメージをアップグレードする

この例では、second_disk ブート環境をアップグレードします。インストールイメージは、2枚のCDに入っています。Oracle Solaris Software - 1 CD と Oracle Solaris Software - 2 CD です。-u オプションでは、CD セットのすべてのパッケージを格納するだけの領域があるかどうかを判定します。-O オプションと -nodisplay および -noconsole オプションを指定すると、2枚目のCDの読み取りの後にキャラクターユーザーインタフェースは表示されません。これらのオプションを使用すると、情報の入力を求められません。

-O オプションに -nodisplay および -noconsole オプションを指定しないと、キャラクターユーザーインタフェース (CUI) が表示されます。Oracle Live Upgrade のタスクを実行するために CUI を使用する必要はありません。

1. アップブレード先のリリースの Live Upgrade パッケージをインストールします。

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

2. Oracle Solaris Software - 1 CD を挿入し、次のコマンドを入力します。

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


3. Oracle Solaris Software - 2 CD を挿入し、次のコマンドを入力します。

```
# luupgrade -i -n second_disk -s /cdrom/cdrom0 -O '-nodisplay \
-noconsole'
Repeat this step for each CD that you need.
```

4. インストールする各 CD について、上記の手順を繰り返します。

次の手順 このようにして、ブート環境をアクティブにする準備が整います。[113 ページ](#)の「ブート環境のアクティブ化」を参照してください。

ブート環境のネットワークインストールイメージに対するパッケージの追加またはパッケージの削除

このセクションでは、新しいブート環境でパッケージを削除したり、パッケージを追加したりする方法について説明します。



注意 -Live Upgrade でアップグレードしたりパッケージまたはパッチの追加や削除を行なったりするには、パッケージまたはパッチが最新の SVR4 パッケージガイドラインに準拠している必要があります。Oracle のパッケージはこのガイドラインに準拠していますが、サードパーティーベンダーのパッケージがこれに準拠していることを Oracle は保証できません。非準拠のパッケージを追加しようとすると、パッケージ追加ソフトウェアの障害が発生するか、アクティブブート環境が変更されてしまう可能性があります。

パッケージの要件の詳細については、[付録 C 「その他の SVR4 パッケージ要件 \(リファレンス\)」](#) を参照してください。

新しいブート環境から 1 つのパッケージまたは一連のパッケージを削除するには、**-P** オプションを使用します。

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

-P 指定した 1 つまたは複数のパッケージをブート環境から削除することを示します。

-n BE-name パッケージを削除するブート環境の名前を指定します。

package-name 削除するパッケージの名前を指定します。複数のパッケージ名を指定する場合は、スペースで区切ります。

新しいブート環境に 1 つのパッケージまたは一連のパッケージを追加するには、**-P** オプションを使用します。

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

-p ブート環境にパッケージを追加することを示します。

-n BE-name パッケージを追加するブート環境の名前を指定します。

-s path-to-packages 追加するパッケージが含まれているディレクトリへのパスを指定します。

package-name 追加するパッケージの名前を指定します。複数のパッケージ名を指定する場合は、スペースで区切ります。

例 5-4 ブート環境のネットワークインストールイメージへのパッケージの追加

この例では、second_disk ブート環境でパッケージを削除し、次に追加します。

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

ブート環境のネットワークインストールイメージに対するパッチの追加またはパッチの削除

このセクションでは、新しいブート環境でパッチを削除したり、パッチを追加したりする方法について説明します。



注意 - Live Upgrade でパッケージまたはパッチの追加や削除を行うには、パッケージまたはパッチが最新の SVR4 パッケージガイドラインに準拠している必要があります。Oracle のパッケージはこのガイドラインに準拠していますが、サードパーティーベンダーのパッケージがこれに準拠していることを Oracle は保証できません。非準拠のパッケージを追加しようとすると、パッケージ追加ソフトウェアの障害が発生するか、アクティブブート環境が改変されてしまう可能性があります。

アクティブなブート環境で Solaris 8 または Solaris 9 OS を実行している場合、Live Upgrade を使用して Oracle Solaris 10 の非アクティブブート環境にパッチを適用することはできません。Live Upgrade は、アクティブなブートパーティション上でパッチユーティリティを呼び出して非アクティブブートパーティションにパッチを適用します。Solaris 8 および Solaris 9 のパッチユーティリティは、Oracle Solaris ゾーン、サービス管理機能 (SMF)、および Oracle Solaris 10 OS のその他の拡張機能を認識しません。そのため、パッチユーティリティは非アクティブな Oracle Solaris 10 ブート環境にパッチを正しく適用できません。

Live Upgrade を使用してシステムを Solaris 8 または Solaris 9 OS から Oracle Solaris 10 OS にアップグレードする場合は、パッチを適用する前に、Oracle Solaris 10 ブート環境

をまずアクティブにする必要があります。Oracle Solaris 10 ブート環境をアクティブにしたら、このアクティブなブート環境にパッチを直接適用できます。または、別の非アクティブブート環境を設定し、Live Upgrade を使用してそのブート環境にパッチを適用することもできます。Solaris 8 から Oracle Solaris 10 リリースへのアップグレードとパッチ適用の例については、[Solaris Live Upgrade の使用の制限事項](#)を参照してください。

新しいブート環境から 1 つのパッチまたは一連のパッチを削除するには、`-T` オプションを使用します。

```
# luupgrade -T -n BE-name patch_name
```

`-T` 指定したパッチをブート環境から削除することを示します。

`-n BE-name` パッチを削除するブート環境の名前を指定します。

`patch-name` 削除するパッチの名前を指定します。複数のパッチ名を指定する場合は、スペースで区切ります。

新しいブート環境に 1 つのパッチまたは一連のパッチを追加するには、`-t` オプションを使用します。

```
# luupgrade -t -n BE-name -s /path-to-patches patch_name
```

`-t` ブート環境にパッチを追加することを示します。

`-n BE-name` パッチを追加するブート環境の名前を指定します。

`-s path-to-patches` 追加するパッチが含まれているディレクトリへのパスを指定します。

`patch-name` 追加するパッチの名前を指定します。複数のパッチ名を指定する場合は、スペースで区切ります。

例 5-5 ブート環境のネットワークインストールイメージにパッチを追加する

この例では、`second_disk` ブート環境に対してパッチの削除と追加を行います。

```
# luupgrade -T -n second_disk 22222-01
```

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

ブート環境にインストールされているパッケージの確認

新しいブート環境に新しくインストールされたパッケージの整合性を確認するには、次のように `pkgchk` コマンドを実行します。

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

-C 指定したパッケージに対して pkgchk コマンドを実行することを示します。

-n BE-name 検査を実行するブート環境の名前を指定します。

-O pkgchk コマンドにオプションを直接渡します。

package-name 検査するパッケージの名前を指定します。複数のパッケージ名を指定する場合は、スペースで区切ります。パッケージ名を省略すると、指定したブート環境にあるすべてのパッケージに対して検査が実行されます。

“-v” コマンドを冗長モードで実行することを指定します。

例 5-6 ブート環境のパッケージの整合性を確認する

この例では、パッケージ SUNWabc、SUNWdef、および SUNWghi を検査して、これらが正しくインストールされ損傷がないことを確認します。

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

JumpStart プロファイルを使用したアップグレード

Live Upgrade で使用する JumpStart プロファイルを作成できます。次の手順により、luupgrade コマンドに -j オプションを指定して使用して、プロファイルの作成、プロファイルのテスト、およびインストールを行うことができます。



注意 - Oracle Solaris OS をフラッシュアーカイブによりインストールする場合は、アーカイブおよびインストールメディアに同一の OS バージョンが含まれている必要があります。たとえば、アーカイブが Oracle Solaris 10 オペレーティングシステムで、DVD メディアを使用する場合は、Solaris 10 DVD メディアを使用してアーカイブをインストールする必要があります。OS バージョンが一致しないと、ターゲットシステムへのインストールは失敗します。次のキーワードまたはコマンドを使用する場合は、同一のオペレーティングシステムが必要です。

- プロファイルの archive_location キーワード
- -s、-a、-j、および -J オプションを指定した luupgrade コマンド

詳細については、以下を参照してください。

- [101 ページの「Live Upgrade で使用されるプロファイルを作成する方法」](#)
- [102 ページの「Live Upgrade で使用されるプロファイルをテストする方法」](#)

- 103 ページの「Live Upgrade を使用してプロファイルでアップグレードする方法」
- JumpStart プロファイルの作成については、『Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール』の「プロファイルの作成」を参照してください。

▼ Live Upgrade で使用されるプロファイルを作成する方法

この手順は、Live Upgrade で使用されるプロファイルを作成する方法を示しています。luupgrade コマンドに -j オプションを指定してこのプロファイルを使用し、非アクティブブート環境をアップグレードできます。

このプロファイルの使用方法については、次のセクションを参照してください。

- プロファイルを使用したアップグレードについては、103 ページの「Live Upgrade を使用してプロファイルでアップグレードする方法」を参照してください。
- プロファイルを使用したフラッシュアーカイブインストールについては、110 ページの「プロファイルを使用してフラッシュアーカイブをインストールする方法」を参照してください。

1 テキストエディタを使用してテキストファイルを作成します。

ファイルにわかりやすい名前を付けます。プロファイルの名前は、システムに Oracle Solaris ソフトウェアをインストールするためにそのプロファイルをどのように使用するかを示すものにしてください。たとえば、このプロファイルに upgrade_Solaris_10 という名前を付けます。

2 プロファイルにプロファイルキーワードと値を追加します。

表 5-3 および表 5-4 のアップグレードキーワードのみを Live Upgrade プロファイルで使用できます。

次の表に、Install_type キーワードの値 upgrade または flash_install とともに使用できるキーワードを示します。

3 プロファイルをローカルシステムのディレクトリに保存します。

4 プロファイルの所有者が root で、そのアクセス権が 644 に設定されていることを確認します。

例 5-7 Live Upgrade プロファイルの作成

この例では、プロファイルはアップグレードパラメータを提供します。このプロファイルを使用して、Live Upgrade の luupgrade コマンドに -u および -j オプションを指定して、非アクティブなブート環境をアップグレードします。このプロファイルでは、パッケージおよびクラスタを追加します。地域ロケールおよび追加ロ

ケールも、プロファイルに追加されます。プロファイルにロケールを追加する場合、ブート環境がディスク容量に余裕を持って作成されていることを確認してください。

# profile keywords	profile values
# -----	-----
install_type	upgrade
package	SUNWxwman 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

例 5-8 差分アーカイブをインストールするための Live Upgrade プロファイルの作成

次のプロファイルの例は、Live Upgrade でクローンシステムに差分アーカイブをインストールするためのものです。差分アーカイブで指定されているファイルだけが、追加、削除、または変更されます。フラッシュアーカイブは、NFS サーバーから取得されます。イメージは元のマスターシステムから構築されたものなので、クローンシステムのイメージの妥当性検査は行われません。このプロファイルは、Live Upgrade の `luupgrade` コマンドの `-u` オプションと `-j` オプションで使します。

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

`luupgrade` コマンドを使用して差分アーカイブをインストールする方法については、[110 ページの「プロファイルを使用してフラッシュアーカイブをインストールする方法」](#)を参照してください。

次の手順 プロファイルをテストする場合は、[102 ページの「Live Upgrade で使用されるプロファイルをテストする方法」](#)を参照してください。

▼ Live Upgrade で使用されるプロファイルをテストする方法

プロファイルを作成したら、`luupgrade` コマンドを使用してプロファイルをテストします。`luupgrade` が生成するインストール出力を調べることによって、意図した通りにプロファイルが動作するかをすばやく調べることができます。

- プロファイルをテストします。

```
# luupgrade -u -n BE-name -D -s os-image-path -j profile-path
```

-u	ブート環境のオペレーティングシステムイメージをアップグレードします。
-n BE-name	アップグレード対象のブート環境の名前を指定します。
-D	選択されたブート環境のディスク構成を使用して、-j オプションで渡されたプロファイルオプションをテストします。
-s os-image-path	オペレーティングシステムイメージが置かれているディレクトリのパス名を指定します。このディレクトリはインストールメディア (DVD-ROM や CD-ROM など) 上でも NFS または UFS ディレクトリであってもかまいません。
-j profile-path	アップグレード用に構成されたプロファイルのパス。プロファイルは、ローカルマシンのディレクトリに存在する必要があります。

例 5-9 Live Upgrade を使用したプロファイルのテスト

次の例では、プロファイル名は `Flash_profile` です。このプロファイルは、`second_disk` という名前の非アクティブブート環境でのテストに成功します。

```
# luupgrade -u -n ulb08 -D -s /net/installsvr/export/u1/combined.ulwos \
-j /var/tmp/flash_profile
Validating the contents of the media /net/installsvr/export/u1/combined.ulwos.
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.
```

これで、プロファイルを使用して非アクティブブート環境をアップグレードできます。

▼ Live Upgrade を使用してプロファイルでアップグレードする方法

ここでは、プロファイルを使用した OS のアップグレード方法を、手順を追って説明します。

プロファイルを使用してフラッシュアーカイブをインストールする場合は、[110 ページの「プロファイルを使用してフラッシュアーカイブをインストールする方法」](#)を参照してください。

始める前に プロファイルにロケールを追加した場合、ブート環境がディスク容量に余裕を持って作成されていることを確認してください。



注意 - Oracle Solaris OS をフラッシュアーカイブによりインストールする場合は、アーカイブおよびインストールメディアに同一の OS バージョンが含まれている必要があります。たとえば、アーカイブが Oracle Solaris 10 オペレーティングシステムで、DVD メディアを使用する場合は、Solaris 10 DVD メディアを使用してアーカイブをインストールする必要があります。OS バージョンが一致しないと、ターゲットシステムへのインストールは失敗します。次のキーワードまたはコマンドを使用する場合は、同一のオペレーティングシステムが必要です。

- プロファイルの `archive_location` キーワード
- `-s`、`-a`、`-j`、および `-J` オプションを指定した `luupgrade` コマンド

- 1 **Live Upgrade** のパッケージ `SUNWlucfg`、`SUNWlur`、および `SUNWluu` をシステムにインストールします。

これらのパッケージは、アップグレード先のリリースのものである必要があります。手順については、[63 ページの「pkgadd コマンドを使用して Live Upgrade をインストールする方法」](#)を参照してください。

- 2 プロファイルを作成します。

Live Upgrade プロファイルで使用可能なアップグレードキーワードのリストについては、[101 ページの「Live Upgrade で使用されるプロファイルを作成する方法」](#)を参照してください。

- 3 プロファイルを使用して OS をアップグレードします。

```
# luupgrade -u -n BE-name -s os-image-path -j profile-path
```

- | | |
|-------------------------------|--|
| <code>-u</code> | ブート環境のオペレーティングシステムイメージをアップグレードします。 |
| <code>-n BE-name</code> | アップグレード対象のブート環境の名前を指定します。 |
| <code>-s os-image-path</code> | オペレーティングシステムイメージが置かれているディレクトリのパス名を指定します。このディレクトリはインストールメディア (DVD-ROM や CD-ROM など) 上でも NFS または UFS ディレクトリであってもかまいません。 |
| <code>-j profile-path</code> | プロファイルへのパス。プロファイルは、ローカルマシンのディレクトリに存在する必要があります。プロファイルの作成に |

については、101 ページの「Live Upgrade で使用されるプロファイルを作成する方法」を参照してください。

例 5-10 カスタム JumpStart プロファイルを使用してブート環境をアップグレードする

この例では、second_disk ブート環境はプロファイルを使用してアップグレードされます。-j オプションを使用して、プロファイルにアクセスします。このようにして、ブート環境をアクティブにする準備が整います。プロファイルを作成する方法については、101 ページの「Live Upgrade で使用されるプロファイルを作成する方法」を参照してください。pkgadd コマンドは、アップグレード先のリリースの Live Upgrade パッケージを追加します。

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

次の手順 このようにして、ブート環境をアクティブにする準備が整います。113 ページの「ブート環境のアクティブ化」を参照してください。

JumpStart キーワード値

表 5-3 初期アーカイブ作成のキーワード

キーワード	説明	リファレンス
(必須) Install_type	システムの既存の Oracle Solaris 環境をアップグレードするか、システムにフラッシュアーカイブをインストールするかを定義します。このキーワードには、次の値を使用します。 <ul style="list-style-type: none">■ アップグレードの場合は、upgrade■ フラッシュアーカイブのインストールの場合は、flash_install■ フラッシュアーカイブの差分インストールの場合は、flash_update	このキーワードのすべての値の説明は、『Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール』の「install_type プロファイルキーワード (UFS と ZFS)」を参照してください。

表 5-3 初期アーカイブ作成のキーワード (続き)

キーワード	説明	リファレンス
(フラッシュアーカイブの場合は必須) archive_location	指定された位置からフラッシュアーカイブを取得します。	このキーワードで使用可能な値のリストについては、『 Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール 』の「 archive_location キーワード 」を参照してください。
(オプション) cluster (クラスタの追加または削除)	システムにインストールされるソフトウェアグループに対してクラスタを追加するか、または削除するかを指定します。	このキーワードで使用可能な値のリストについては、『 Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール 』の「 cluster プロファイルキーワード (ソフトウェアグループの追加) (UFS と ZFS) 」を参照してください。
(オプション) geo	システムにインストールする地域ロケールか、あるいはシステムのアップグレード時に追加する地域ロケールを指定します。	このキーワードで使用可能な値のリストについては、『 Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール 』の「 geo プロファイルキーワード (UFS と ZFS) 」を参照してください。
(オプション) local_customization	クローンシステムにフラッシュアーカイブをインストールする前に、カスタムスクリプトを作成して、クローンシステム上にローカル構成を保持できます。local_customization キーワードは、これらのスクリプトの格納先ディレクトリを示します。この値は、クローンシステム上のスクリプトへのパスです。	配備前および配備後スクリプトについては、『 Oracle Solaris 10 1/13 インストールガイド: フラッシュアーカイブ (作成とインストール) 』の「 カスタマイズスクリプトの作成 」を参照してください。
(オプション) locale	インストールまたはアップグレード時に追加するロケールパッケージを指定します。	このキーワードで使用可能な値のリストについては、『 Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール 』の「 locale プロファイルキーワード (UFS と ZFS) 」を参照してください。

表 5-3 初期アーカイブ作成のキーワード (続き)

キーワード	説明	リファレンス
(オプション) <code>package</code>	システムにインストールされるソフトウェアグループに対してパッケージを追加するか、または削除するかを指定します。	このキーワードで使用可能な値のリストについては、『 Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール 』の「 <code>package</code> プロファイルキーワード (UFS と ZFS)」を参照してください。

表 5-4 差分アーカイブ作成のキーワード

キーワード	説明	リファレンス
(必須) <code>Install_type</code>	システムにフラッシュアーカイブをインストールするようにインストールを定義します。差分アーカイブを表す値は <code>flash_update</code> です。	このキーワードのすべての値の説明は、『 Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール 』の「 <code>install_type</code> プロファイルキーワード (UFS と ZFS)」を参照してください。
(必須) <code>archive_location</code>	指定された位置からフラッシュアーカイブを取得します。	このキーワードで使用可能な値のリストについては、『 Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール 』の「 <code>archive_location</code> キーワード」を参照してください。
(オプション) <code>forced_deployment</code>	ソフトウェアで想定されているものとは異なるクローンシステムに、フラッシュアーカイブ差分アーカイブを強制的にインストールします。 <code>forced_deployment</code> を使用すると、クローンシステムを期待される状態にするために、新規ファイルがすべて削除されます。ファイルを削除して良いかどうか判断できない場合には、デフォルトを使用してください。デフォルトでは、新規ファイルが削除されそうになると、インストールが停止します。	このキーワードの詳細は、『 Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール 』の「 <code>forced_deployment</code> プロファイルキーワード (フラッシュアーカイブ差分アーカイブのインストール)」を参照してください。

表 5-4 差分アーカイブ作成のキーワード (続き)

キーワード	説明	リファレンス
(オプション) local_customization	クローンシステムにフラッシュアーカイブをインストールする前に、カスタムスクリプトを作成して、クローンシステム上にローカル構成を保持できます。local_customization キーワードは、これらのスクリプトの格納先ディレクトリを示します。この値は、クローンシステム上のスクリプトへのパスです。	配備前および配備後スクリプトについては、『 Oracle Solaris 10 1/13 インストールガイド: フラッシュアーカイブ (作成とインストール)』 の「カスタマイズスクリプトの作成」を参照してください。
(オプション) no_content_check	フラッシュアーカイブ差分アーカイブを使用してクローンシステムをインストールする場合、no_content_check キーワードを使用してファイルごとの検証を省略できます。ファイルごとの検証により、クローンシステムがマスターシステムの複製であることが保証されます。クローンシステムが元のマスターシステムの複製であることが確実である場合を除き、このキーワードの使用は避けてください。	このキーワードの詳細は、『 Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール』 の「no_content_check プロファイルキーワード (フラッシュアーカイブのインストール)」を参照してください。
(オプション) no_master_check	フラッシュアーカイブ差分アーカイブを使用してクローンシステムをインストールする場合、no_master_check キーワードを使用してファイルの検証を省略できます。クローンシステムのファイルの検証は行われません。この検証により、クローンシステムが元のマスターシステムから構築されていることが保証されます。クローンシステムが元のマスターシステムの複製であることが確実である場合を除き、このキーワードの使用は避けてください。	このキーワードの詳細は、『 Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール』 の「no_master_check プロファイルキーワード (フラッシュアーカイブのインストール)」を参照してください。

ブート環境へのフラッシュアーカイブのインストール

このセクションでは、Live Upgrade を使用してフラッシュアーカイブをインストールする手順を示します。フラッシュアーカイブをインストールすると、新しいブート環境上にある共有ファイル以外のすべてのファイルが上書きされます。アーカイブは、次のメディアに格納されています。

- HTTP サーバー
- FTP サーバー - コマンド行インタフェースでのみ使用可能
- NFS サーバー
- ローカルファイル
- ローカルテープ
- ローカルデバイス (DVD または CD)

フラッシュアーカイブをインストールおよび作成する場合は次の点に注意してください。

Oracle Solaris OS をフラッシュアーカイブによりインストールする場合は、アーカイブおよびインストールメディアに同一の OS バージョンが含まれている必要があります。OS バージョンが一致しないと、ターゲットシステムへのインストールは失敗します。次のキーワードまたはコマンドを使用する場合は、同一のオペレーティングシステムが必要です。

- プロファイルの `archive_location` キーワード
- `-s`、`-a`、`-j`、および `-J` オプションを指定した `luupgrade` コマンド

非大域ゾーンがインストールされていると、フラッシュアーカイブは正常に作成されません。Solaris フラッシュ機能は、Oracle Solaris ゾーン機能とは互換性がありません。非大域ゾーンにフラッシュアーカイブを作成するか、非大域ゾーンがインストールされている大域ゾーンにアーカイブを作成すると、作成されるアーカイブは、アーカイブが配備されるときに適切にインストールされません。たとえば、アーカイブが Oracle Solaris 10 オペレーティングシステムで、DVD メディアを使用する場合は、Solaris 10 DVD メディアを使用してアーカイブをインストールする必要があります。

アーカイブストレージに関連付けられるパスの正しい構文の例については、『[Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール](#)』の「`archive_location` キーワード」を参照してください。

フラッシュアーカイブインストール機能を使用するには、マスターシステムをインストールし、フラッシュアーカイブを作成します。アーカイブの作成についての詳細は、『[Oracle Solaris 10 1/13 インストールガイド: フラッシュアーカイブ \(作成とインストール\)](#)』の第3章「フラッシュアーカイブの作成 (タスク)」を参照してください。

▼ ブート環境にフラッシュアーカイブをインストールする方法

- 1 Live Upgrade のパッケージ `SUNWlucfg`、`SUNWlur`、および `SUNWluu` をシステムにインストールします。

これらのパッケージは、アップグレード先のリリースのものである必要があります。手順については、[63 ページの「pkgadd コマンドを使用して Live Upgrade をインストールする方法」](#)を参照してください。

- 2 フラッシュアーカイブからインストールします。

```
# luupgrade -f -n BE-name -s os-image-path -a archive
```

`-f` オペレーティングシステムをフラッシュアーカイブからインストールすることを示します。

<code>-n BE-name</code>	アーカイブを使用してインストールするブート環境の名前を指定します。
<code>-s os-image-path</code>	オペレーティングシステムイメージが置かれているディレクトリのパス名を指定します。このディレクトリはインストールメディア (DVD-ROM や CD-ROM など) 上でも NFS または UFS ディレクトリであってもかまいません。この OS イメージでは、最低限のブート可能ルート (/) ファイルシステムをブートするミニルートが提供され、フラッシュアーカイブを簡単にインストールできます。ミニルートは、インストールされているイメージではありません。 <code>-a</code> オプションは、オペレーティングシステムイメージを指定します。
<code>-a archive</code>	フラッシュアーカイブへのパス (ローカルファイルシステムでそのアーカイブが利用できる場合)。 <code>-s</code> オプションと <code>-a</code> オプションを使用して指定するオペレーティングシステムイメージのバージョンは、同じでなければなりません。

例 5-11 ブート環境へのフラッシュアーカイブのインストール

この例では、アーカイブは `second_disk` ブート環境にインストールされます。アーカイブはローカルシステムに存在します。`-a` オプションでは、オペレーティングシステムのイメージが提供されます。`-s` および `-a` オプションのオペレーティングシステムバージョンは、どちらも Oracle Solaris 10 8/11 リリースです。`second_disk` 上のファイルは、共有可能ファイルを除いてすべて上書きされます。`pkgadd` コマンドは、アップグレード先のリリースの Live Upgrade パッケージを追加します。

```
# 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
```

次の手順 このようにして、ブート環境をアクティブにする準備が整います。[113 ページ](#)の「ブート環境のアクティブ化」を参照してください。

▼ プロファイルを使用してフラッシュアーカイブをインストールする方法

この手順は、プロファイルを使用してフラッシュアーカイブまたは差分アーカイブをインストールする手順を示しています。

始める前に プロファイルにロケールを追加した場合、ブート環境がディスク容量に余裕を持って作成されていることを確認してください。

- 1 **Live Upgrade** のパッケージ **SUNWlucfg**、**SUNWlur**、および **SUNWluu** をシステムにインストールします。

これらのパッケージは、アップグレード先のリリースのものである必要があります。手順については、[63 ページの「pkgadd コマンドを使用して Live Upgrade をインストールする方法」](#)を参照してください。

- 2 プロファイルを作成します。

Live Upgrade プロファイルで使用可能なキーワードのリストについては、[101 ページの「Live Upgrade で使用されるプロファイルを作成する方法」](#)を参照してください。

- 3 プロファイルを使用してフラッシュアーカイブをインストールします。

```
# luupgrade -f -n BE-name -s os-image-path -j profile-path
```

-f	オペレーティングシステムをフラッシュアーカイブからインストールすることを示します。
-n BE-name	アップグレード対象のブート環境の名前を指定します。
-s os-image-path	オペレーティングシステムイメージが置かれているディレクトリのパス名を指定します。このディレクトリはインストールメディア (DVD-ROM や CD-ROM など) 上でも NFS または UFS ディレクトリであってもかまいません。この OS イメージでは、最低限のブート可能ルート (/) ファイルシステムをブートするミニルートが提供され、フラッシュアーカイブを簡単にインストールできます。ミニルートは、インストールされているイメージではありません。-j オプションは、フラッシュアーカイブのオペレーティングシステムイメージを含むプロファイルへのパスを指定します。
-j profile-path	フラッシュインストール用に構成された JumpStart プロファイルへのパス。プロファイルは、ローカルマシンのディレクトリに存在する必要があります。-s オプションのオペレーティングシステムのバージョンとフラッシュアーカイブのオペレーティングシステムのバージョンは、同一でなければいけません。

例 5-12 プロファイルを使用してブート環境にフラッシュアーカイブをインストールする

この例では、インストールするアーカイブの場所をプロファイルで指定しています。

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

プロファイルを作成した後、luupgrade コマンドを実行してアーカイブをインストールできます。-s オプションは、最低限のブート可能ルート (/) ファイルシステムをブートするミニルートを指定し、フラッシュアーカイブを簡単にインストールできます。-j オプションは、フラッシュアーカイブのオペレーティングシステムイメージへのパスを含むプロファイルへのパスを指定します。pkgadd コマンドは、アップグレード先のリリースの Live Upgrade パッケージを追加します。

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

次の手順 このようにして、ブート環境をアクティブにする準備が整います。113 ページの「ブート環境のアクティブ化」を参照してください。

▼ プロファイルキーワードを使用してフラッシュアーカイブをインストールする方法

この手順により、フラッシュアーカイブをインストールし、プロファイルファイルでなくコマンド行から archive_location キーワードを使用できます。プロファイルファイルを使用せずに、アーカイブを取得できます。

- 1 Live Upgrade のパッケージ SUNWlucfg、SUNWlur、および SUNWluu をシステムにインストールします。

これらのパッケージは、アップグレード先のリリースのものである必要があります。手順については、63 ページの「pkgadd コマンドを使用して Live Upgrade をインストールする方法」を参照してください。

- 2 フラッシュアーカイブをインストールします。

```
# luupgrade -f -n BE-name -s os-image-path -J 'archive_location path-to-profile'
```

-f	オペレーティングシステムをフラッシュアーカイブからアップグレードすることを指定します。
-n BE-name	アップグレード対象のブート環境の名前を指定します。
-s os-image-path	オペレーティングシステムイメージが置かれているディレクトリのパス名を指定します。このディレクトリはインストールメディア (DVD-ROM や CD-ROM など) 上でも NFS または UFS ディレクトリであってもかまいません。この OS イメージでは、最低限のブート可能ルート (/) ファイルシステムをブートするミニルートが提供

され、フラッシュアーカイブを簡単にインストールできます。ミニルートは、インストールされているイメージではありません。-j オプションは、フラッシュアーカイブのオペレーティングシステムイメージを含むプロファイルへのパスを指定します。

-J 'archive_location path-to-profile'

archive_location プロファイルキーワードと、JumpStart プロファイルへのパスを指定します。-s オプションのオペレーティングシステムのバージョンとフラッシュアーカイブのオペレーティングシステムのバージョンは、同一でなければいけません。キーワードの値については、『Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール』の「archive_location キーワード」を参照してください。

このようにして、ブート環境をアクティブにする準備が整います。113 ページの「ブート環境のアクティブ化」を参照してください。

例 5-13 プロファイルキーワードを使用してフラッシュアーカイブをインストールする

この例では、アーカイブは second_disk ブート環境にインストールされます。-s オプションは、最低限のブート可能ルート (/) ファイルシステムをブートするミニルートを指定し、フラッシュアーカイブを簡単にインストールできます。-j オプションは、フラッシュアーカイブのオペレーティングシステムイメージへのパスを指定します。-J オプションと archive_location キーワードを使用して、アーカイブを取得します。second_disk 上のファイルは、共有可能ファイルを除いてすべて上書きされます。pkgadd コマンドは、アップグレード先のリリースの Live Upgrade パッケージを追加します。

```
# 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'
```

ブート環境のアクティブ化

ブート環境をアクティブにすると、次のシステムリブートでブート可能になります。新しいアクティブブート環境で何か問題が発生する場合は、元のブート環境にすぐに戻すことができます。第 6 章「障害回復: 元のブート環境へのフォールバック (タスク)」を参照してください。

- `luactivate` コマンドによるブート環境のアクティブ化については、[115 ページ](#)の「ブート環境をアクティブにする方法」を参照してください。

注-ブート環境をはじめてアクティブにする場合は、`luactivate` コマンドを使用する必要があります。

- ブート環境のアクティブ化と強制的なファイルの同期については、[116 ページ](#)の「ブート環境をアクティブにしてファイルを同期させる方法」を参照してください。

注-最初のアクティブ化の際にファイルが同期されます。最初のアクティブ化のあとでブート環境を切り替える場合、ファイルは同期されません。

- Oracle Solaris の機能である x86 上の GRUB メニューを使用したブート環境のアクティブ化については、[119 ページ](#)の「x86: GRUB メニューを使ってブート環境をアクティブにする方法」を参照してください。

注-GRUB メニューを使用すると、ブート環境を簡単に切り替えることができます。最初のアクティブ化のあと、ブート環境が GRUB メニューに表示されます。

ブート環境をアクティブ化するための要件と制限

ブート環境を正常にアクティブにするためには、そのブート環境が次の条件を満たしている必要があります。

- ブート環境のステータスは `Complete` である必要があります。ステータスを確認する方法については、[132 ページ](#)の「すべてのブート環境のステータスの表示」を参照してください。
- ブート環境が現在のブート環境でない場合は、`luumount` または `mount` コマンドを使用してそのブート環境のパーティションをマウントすることはできません。[luumount\(1M\)](#) または [mount\(1M\)](#) のマニュアルページを参照してください。
- 比較処理で使用中のブート環境はアクティブにできません。手順については、[135 ページ](#)の「ブート環境の比較」を参照してください。スワップを再構成するには、[73 ページ](#)の「ブート環境を作成しスワップを再構成する方法」を参照してください。
- スワップを再構成する場合は、非アクティブブート環境をブートする前に実行してください。デフォルトでは、すべてのブート環境が同じスワップデバイスを共有します。

x86 のみ - x86 システムの場合は、GRUB メニューを使用してアクティブにすることもできます。次の例外について注意してください。

- ブート環境が Solaris 8、Solaris 9、または Solaris 10 3/05 リリースで作成されている場合、ブート環境は常に `luactivate` コマンドでアクティブにする必要があります。これらの古いブート環境は、GRUB メニューには表示されません。
- ブート環境をはじめてアクティブにする場合は、`luactivate` コマンドを使用する必要があります。次のブート時には、そのブート環境の名前が GRUB メインメニューに表示されます。それにより、GRUB メニューで適切なエントリを選択して、このブート環境に切り替えることができます。

117 ページの「[x86: GRUB メニューを使ったブート環境のアクティブ化](#)」を参照してください。

▼ ブート環境をアクティブにする方法

次の手順は、新しいブート環境を切り替えて現在稼働中のブート環境にします。

x86 のみ - x86 システムの場合は、GRUB メニューを使用してアクティブにすることもできます。次の例外について注意してください。

- ブート環境が Solaris 8、Solaris 9、または Solaris 10 3/05 リリースで作成されている場合、ブート環境は常に `luactivate` コマンドでアクティブにする必要があります。これらの古いブート環境は、GRUB メニューには表示されません。
- ブート環境をはじめてアクティブにする場合は、`luactivate` コマンドを使用する必要があります。次のブート時には、そのブート環境の名前が GRUB メインメニューに表示されます。それにより、GRUB メニューで適切なエントリを選択して、このブート環境に切り替えることができます。

117 ページの「[x86: GRUB メニューを使ったブート環境のアクティブ化](#)」を参照してください。

- 1 ブート環境をアクティブにするには、次のように入力します。

```
# /sbin/luactivate BE-name
```

`BE-name` アクティブにするブート環境の名前を指定します。

- 2 リブートします。

```
# init 6
```



注意 - リブートには `init` か `shutdown` コマンドを使用してください。 `reboot` や `halt`、`uadmin` コマンドを使用すると、ブート環境の切り替えは行われません。以前にアクティブであったブート環境が再びブートされます。

例 5-14 ブート環境のアクティブ化

この例では、次のリブート時に `second_disk` ブート環境がアクティブになります。

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

▼ ブート環境をアクティブにしてファイルを同期させる方法

新しく作成したブート環境ではじめてブートするときに、Live Upgrade ソフトウェアは新しいブート環境と以前のアクティブブート環境の同期をとります。「同期」とは、いくつかのクリティカルなシステムファイルおよびディレクトリを、以前にアクティブだったブート環境から、ブート中のブート環境にコピーすることです。Live Upgrade では、強制的に同期を行うように `luactivate` コマンドと `-s` オプションで指定しないかぎり、初期ブート後にこの同期が行われることはありません。

x86 のみ - GRUB メニューを使ってブート環境間で切り替えを行なった場合も、ファイルは同期されません。ファイルを同期させるには、次の手順を使用する必要があります。

同期の詳細については、[55 ページの「ブート環境間でのファイルの同期」](#)を参照してください。

- 1 ブート環境をアクティブにするには、次のように入力します。

```
# /sbin/luactivate -s BE-name
```

`-s` 以前にアクティブであったブート環境と新しいブート環境のファイルを同期します。ブート環境の最初のアクティブ化ではファイル間の同期が行われますが、それ以降のアクティブ化では `-s` オプションを指定しない限りファイルの同期は行われません。



注意-以前のアクティブブート環境で発生した変更によりユーザーが気付いていない場合や、それらの変更を制御できない場合もあるため、このオプションを使用する際には十分注意してください。たとえば、現在のブート環境で Oracle Solaris 10 8/11 ソフトウェアを実行していて、強制的に同期をとってブート環境を Solaris 9 リリースに戻したとします。この場合、Solaris 9 リリースでファイルが変更されることがあります。ファイルは OS のリリースに依存しているため、Solaris 9 リリースのブートは失敗する可能性があります。Solaris Oracle Solaris 10 8/11 のファイルと Solaris 9 のファイルは互換性があるとは限らないからです。

BE-name アクティブにするブート環境の名前を指定します。

2 リブートします。

```
# init 6
```

例 5-15 ブート環境のアクティブ化

この例では、次のリブート時に `second_disk` ブート環境がアクティブになり、ファイルの同期がとられます。

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

x86: GRUB メニューを使ったブート環境のアクティブ化

GRUB メニューでもブート環境間での切り替えを行うことができます。GRUB メニューは、`luactivate` コマンドによるアクティブ化(ブート)に代わるものです。注意事項および制限事項は次のとおりです。



注意-ブート環境をアクティブにしたあとは、BIOS ディスク順序を変更しないでください。この順序を変更すると、GRUB メニューが無効になる可能性があります。この問題が発生した場合は、ディスク順序を元の順序に変更すれば、GRUB メニューが修正されます。

- ブート環境をはじめてアクティブにする場合は、`luactivate` コマンドを使用する必要があります。次のブート時には、そのブート環境の名前が GRUB メインメニューに表示されます。それにより、GRUB メニューで適切なエントリを選択し

て、このブート環境に切り替えることができます。ブート環境をアクティブにする方法については、[115 ページの「ブート環境をアクティブにする方法」](#)を参照してください。

- はじめてブート環境をアクティブにすると、現在のブート環境と新しいブート環境の間でファイルが同期されます。以降のアクティブ化では、ファイルは同期されません。GRUB メニューを使ってブート環境間で切り替えを行なった場合も、ファイルは同期されません。-s オプションを指定して `luactivate` コマンドを使用すると、強制的に同期処理を実行できます。ブート環境をアクティブにして、ファイルを同期する方法については、[116 ページの「ブート環境をアクティブにしてファイルを同期させる方法」](#)を参照してください。
- ブート環境が Solaris 8、Solaris 9、または Solaris 10 3/05 リリースで作成されている場合、ブート環境は常に `luactivate` コマンドでアクティブにする必要があります。これらの古いブート環境は、GRUB メニューには表示されません。ブート環境をアクティブにする方法については、[115 ページの「ブート環境をアクティブにする方法」](#)を参照してください。
- `menu.lst` ファイルには、GRUB メニューに表示される情報が含まれています。このファイルは、次の理由のために修正することができます。
 - Oracle Solaris OS 以外のオペレーティングシステムのために GRUB メニューエントリに追加する。詳細は、『[Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画](#)』の「GRUB ベースのブート (概要)」を参照してください。
 - ブート動作をカスタマイズする。たとえば、ブートを詳細モードに変更したり、自動的に OS をブートするデフォルトの時間を変更したりできます。詳細は、『[Oracle Solaris の管理: 基本管理](#)』の「[GRUB を使用して x86 システムをブートする \(タスクマップ\)](#)」を参照してください。

注 - GRUB メニューを変更する場合は、`menu.lst` ファイルを見つける必要があります。手順については、『[Oracle Solaris の管理: 基本管理](#)』の第 13 章「[Oracle Solaris ブートアーカイブの管理 \(タスク\)](#)」を参照してください。



注意 - Live Upgrade エントリを変更する場合は、`GRUBmenu.lst` ファイルを使用しないでください。変更すると、Live Upgrade が失敗する可能性があります。`menu.lst` ファイルを使用してブート動作をカスタマイズすることはできますが、推奨されるカスタマイズの方法は `eeeprom` コマンドを使用することです。`menu.lst` ファイルを使用してカスタマイズすると、ソフトウェアのアップグレード中に Oracle Solaris OS エントリが変更される場合があります。ファイルへの変更は失われる可能性があります。

▼ x86: GRUB メニューを使ってブート環境をアクティブにする方法

GRUB メニューを使って2つのブート環境間での切り替えを行うことができます。次の制限に注意してください。

- ブート環境の最初のアクティブ化は、`luactivate` コマンドを使って行う必要があります。最初のアクティブ化のあと、ブート環境が GRUB メニューに表示されます。そのあと、ブート環境を GRUB メニューからブートすることができます。
- GRUB メニューを使用してブート環境に切り替えると、同期が省略されます。ファイルの同期の詳細は、[56 ページの「ブート環境間での強制的な同期」](#)のリンクを参照してください。
- ブート環境が Solaris 8、Solaris 9、または Solaris 10 3/05 リリースで作成されている場合、ブート環境は常に `luactivate` コマンドでアクティブにする必要があります。これらの古いブート環境は、GRUB メニューには表示されません。

1 システムをリブートします。

```
# init 6
```

GRUB メインメニューが表示されます。

この例では、Oracle Solaris と、Live Upgrade のブート環境である `second_disk` の2つのオペレーティングシステムが表示されます。`failsafe` エントリは、なんらかの理由でプライマリ OS がブートしない場合の回復用です。

```
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.
```

- ### 2 矢印キーを使用して目的のブート環境を選択し、**Enter**を押します。
- 選択したブート環境がブートされ、アクティブなブート環境になります。

障害回復:元のブート環境へのフォールバック(タスク)

この章では、アクティブ化によって発生した問題を解決する方法について説明します。

注- この章では、UFS ファイルシステム用の Live Upgrade について説明します。luactivate コマンドの使用法は、ZFS ブート環境でも同じです。UFS ファイルシステムを ZFS ルートプールに移行する、または ZFS ルートプールを作成およびインストールする手順については、第 12 章「ZFS ルートプールのブート環境の作成」を参照してください。

アップグレードのあとに障害が検出されたり、アップグレードされたコンポーネントにアプリケーションが対応できない場合は、次のいずれかの手順 (使用しているプラットフォームによって異なる) を使って元のブート環境に戻してください。

- **SPARC システムの場合:**
 - 122 ページの「SPARC: 新しいブート環境のアクティブ化に成功した場合のフォールバック」
 - 122 ページの「SPARC: ブート環境のアクティブ化に失敗した場合のフォールバック」
 - 123 ページの「SPARC: DVD、CD、またはネットワークインストールイメージを使って元のブート環境に戻す」
- **x86 システムの場合:**
 - 124 ページの「x86: 新しいブート環境のアクティブ化に成功した場合の GRUB メニューを使ったフォールバック」
 - 125 ページの「x86: ブート環境のアクティブ化に失敗した場合の GRUB メニューを使ったフォールバック」
 - 127 ページの「x86: ブート環境のアクティブ化に失敗した場合の GRUB メニューと DVD または CD を使ったフォールバック」

SPARC: 元のブート環境へのフォールバック

次のいずれかの方法を使用して、元のブート環境にフォールバックできます。

- 122 ページの「SPARC: 新しいブート環境のアクティブ化に成功した場合のフォールバック」
- 122 ページの「SPARC: ブート環境のアクティブ化に失敗した場合のフォールバック」
- 123 ページの「SPARC: DVD、CD、またはネットワークインストールイメージを使って元のブート環境に戻す」

▼ SPARC: 新しいブート環境のアクティブ化に成功した場合のフォールバック

新しいブート環境のアクティブ化に成功したが、その結果に満足できない場合は、この手順を使用します。

- 1 目的のブート環境をアクティブにします。

```
# /sbin/luactivate BE-name
```

- 2 リブートします。

```
# init 6
```

前の稼働ブート環境がアクティブブート環境になります。

▼ SPARC: ブート環境のアクティブ化に失敗した場合のフォールバック

- 新しいブート環境のブートに失敗した場合で、元のブート環境をシングルユーザーモードでブートできる場合は、この手順を使って元のブート環境に戻します。
- メディアやネットインストールイメージからブートする場合は、123 ページの「SPARC: DVD、CD、またはネットワークインストールイメージを使って元のブート環境に戻す」を参照してください。

- 1 OK プロンプトで、Oracle Solaris Operating System DVD、Oracle Solaris Software - 1 CD、ネットワーク、またはローカルディスクからマシンをシングルユーザーモードでブートします。

```
OK boot device-name -s
```

device-name システムをブートできるデバイスの名前を指定します。/dev/dsk/cwtxdysz という書式を使用します (/dev/dsk/c0t0d0s0 など)。

- 2 目的のブート環境をアクティブにします。

```
# /sbin/luactivate BE-name
```

- このコマンドを実行してもプロンプトが表示されない場合は、[123 ページ](#)の「SPARC: DVD、CD、またはネットワークインストールイメージを使って元のブート環境に戻す」へ進みます。
- プロンプトが表示される場合は、次の手順に進んでください。

- 3 プロンプトで、元のブート環境をアクティブにすることを確認します。

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

フォールバックによるアクティブ化が正常に行われたことを示すメッセージが表示されます。

- 4 リブートします。

```
# init 6
```

前の稼働ブート環境がアクティブブート環境になります。

▼ SPARC: DVD、CD、またはネットワークインストールイメージを使って元のブート環境に戻す

DVD、CD、ネットインストールイメージ、またはブート可能な別のディスクからブートするには、この手順を使用します。この場合、以前のアクティブブート環境からルート (/) スライスをマウントする必要があります。そのあとで、ブート環境の切り替えを行う `luactivate` コマンドを実行してください。リブートすると、以前のアクティブブート環境が再びブートされます。

- 1 OK プロンプトで、**Oracle Solaris Operating System DVD**、**Oracle Solaris Software - 1 CD**、**ネットワーク**、またはローカルディスクからマシンをシングルユーザーモードでブートします。

```
OK boot cdrom -s
```

または

```
OK boot net -s
```

または

```
OK boot device-name -s
```

`device-name` オペレーティングシステムのコピーが格納されているディスクおよびスライスの名前を指定します。`/dev/dsk/cwtxdysz` という書式を使用します (`/dev/dsk/c0t0d0s0` など)。

- 必要に応じて、フォールバックブート環境のルート (/) ファイルシステムの整合性を確認します。

```
# fsck device-name
```

- アクティブブート環境のルート (/) スライスをディレクトリ (/mnt など) にマウントします。

```
# mount device-name /mnt
```

- アクティブなブート環境のルート (/) スライスから、以前の稼働ブート環境をアクティブにします。

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

- ディレクトリをアンマウントします。

```
# umount /mnt
```

- リブートします。

```
# init 6
```

前の稼働ブート環境がアクティブブート環境になります。

x86: 元のブート環境へのフォールバック

元のブート環境に戻すには、ご使用の環境にもっとも適した手順を選択してください。

- 124 ページの「x86: 新しいブート環境のアクティブ化に成功した場合の GRUB メニューを使ったフォールバック」
- 125 ページの「x86: ブート環境のアクティブ化に失敗した場合の GRUB メニューを使ったフォールバック」
- 127 ページの「x86: ブート環境のアクティブ化に失敗した場合の GRUB メニューと DVD または CD を使ったフォールバック」

▼ x86: 新しいブート環境のアクティブ化に成功した場合の **GRUB** メニューを使ったフォールバック

新しいブート環境のアクティブ化に成功したが、その結果に満足できない場合は、この手順を使用します。GRUB を使用すると、元のブート環境にすばやく切り替えることができます。

注- 切り替えるブート環境は、GRUB ソフトウェアで作成された GRUB ブート環境でなければなりません。ブート環境が Solaris 8、Solaris 9、または Solaris 10 3/05 リリースで作成された場合、ブート環境は GRUB ブート環境ではありません。

1 システムをリブートします。

init 6

GRUB メニューが表示されます。Oracle Solaris OS は、元のブート環境です。second_disk ブート環境は正常にアクティブにされていて、GRUB メニューに表示されます。failsafe エントリは、なんらかの理由でプライマリエントリがブートしない場合の回復用です。

```
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.
```

2 元のブート環境をブートするには、矢印キーを使用して元のブート環境を選択し、Return キーを押します。

▼ x86: ブート環境のアクティブ化に失敗した場合の GRUB メニューを使ったフォールバック

ブート中に障害が発生した場合、次の手順を使用して元のブート環境に戻します。この例では、GRUB メニューは適切に表示されますが、新しいブート環境のブートに失敗します。デバイスは /dev/dsk/c0t4d0s0 です。元のブート環境 c0t4d0s0 がアクティブブート環境になります。



注意 - Solaris 10 3/05 リリースでは、以前のブート環境と新しいブート環境が異なるディスク上にある場合に推奨されるフォールバック方法には、BIOS でのハードディスクのブート順序の変更が含まれていました。Solaris 10 1/06 以降のリリースでは、BIOS ディスク順序を変更する必要はなく、変更しないことを強くお勧めします。BIOS ディスク順序を変更すると、GRUB メニューが無効になることがあります。BIOS ディスク順序が変更されている場合は、この順序を元の設定に戻せば、システムの機能が復元されます。

始める前に ブート環境は、GRUB ソフトウェアで作成されている必要があります。**Solaris 10 1/06** リリースより前に作成されたブート環境は、GRUB ブート環境ではありません。ブート可能な GRUB ブート環境がない場合は、[127 ページの「x86: ブート環境のアクティブ化に失敗した場合の GRUB メニューと DVD または CD を使ったフォールバック」](#)の手順に進んでください。

- 1 GRUB メニューを表示するにはシステムをリブートします。

```
# init 6
```

GRUB メニューが表示されます。

```
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.
```

- 2 GRUB メニューから、元のブート環境を選択します。
- 3 GRUB メニューを編集して、シングルユーザーモードでブートします。

- a. **e** と入力して GRUB 編集メニューを表示します。

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

- b. 矢印キーを使用して、元のブート環境のカーネルエントリを選択します。

- c. **e** と入力します。

GRUB 編集メニューにカーネルエントリが表示されます。

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

- d. **-s** と入力し、**Enter** キーを押します。

次の例では、**-s** オプションの配置に注意してください。

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

- e. シングルユーザーモードでブート処理を開始するには、**b** と入力します。

- 4 (省略可能) 必要に応じて、フォールバックブート環境のルート (/) ファイルシステムの整合性を確認します。

```
# fsck mount-point
```

mount-point 信頼性のあるルート (/) ファイルシステム

- 5 元のブート環境のルートスライスをいずれかのディレクトリ (/mnt など) にマウントします。

```
# mount device-name /mnt
```

device-name 戻すブート環境のディスクデバイスにあるルート (/) ファイルシステムの名前を指定します。デバイス名は、/dev/dsk/c wtxd ysz の形式で指定します。

- 6 アクティブなブート環境のルートスライスから、以前の稼働ブート環境をアクティブにします。

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

- 7 ディレクトリをアンマウントします。

```
# umount /mnt
```

- 8 リブートします。

```
# init 6
```

前の稼働ブート環境がアクティブブート環境になります。

▼ x86: ブート環境のアクティブ化に失敗した場合の GRUB メニューと DVD または CD を使ったフォールバック

ブート中に障害が発生した場合、次の手順を使用して元のブート環境に戻します。この例では、新しいブート環境のブートに失敗しています。また、GRUB メニューは表示されません。デバイスは /dev/dsk/c0t4d0s0 です。元のブート環境 c0t4d0s0 がアクティブブート環境になります。



注意 - Solaris 10 3/05 リリースでは、以前のブート環境と新しいブート環境が異なるディスク上にある場合に推奨されるフォールバック方法には、BIOS でのハードディスクのブート順序の変更が含まれていました。Solaris 10 1/06 以降のリリースでは、BIOS ディスク順序を変更する必要はなく、変更しないことを強くお勧めします。BIOS ディスク順序を変更すると、GRUB メニューが無効になることがあります。BIOS ディスク順序が変更されている場合は、この順序を元の設定に戻せば、システムの機能が復元されます。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。

役割には、認証と特権コマンドが含まれます。役割の詳細については、『Solaris のシステム管理: セキュリティーサービス』の「RBAC の構成 (タスクマップ)」を参照してください。

- 2 **Oracle Solaris Operating System DVD (x86 版)** または **Oracle Solaris Software for x86 Platforms -1 CD** を挿入します。

- 3 **DVD** または **CD** からブートします。

```
# init 6
```

GRUB メニューが表示されます。

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

```
+-----+
|Oracle Solaris 10 8/11                               |
|Oracle Solaris 10 8/11 Serial Console ttya           |
|Oracle Solaris 10 8/11 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 デフォルトのオプションがブートされるまで待機するか、表示されているオプションのどれかを選択します。

インストール画面が表示されます。

```
+-----+
|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** オプションを選択します。

次のメッセージが表示されます。

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


- 6 nと入力します。

```
Starting shell...
#
```

これでシングルユーザーモードになりました。

- 7 (省略可能) 必要に応じて、フォールバックブート環境のルート (/) ファイルシステムの整合性を確認します。

```
# fsck mount-point
```

mount-point 信頼性のあるルート (/) ファイルシステム

- 8 元のブート環境のルートスライスをいずれかのディレクトリ (/mnt など) にマウントします。

```
# mount device-name /mnt
```

device-name 戻すブート環境のディスクデバイスにあるルート (/) ファイルシステムの名前を指定します。デバイス名は、/dev/dsk/c wtxd ysz の形式で指定します。

- 9 アクティブなブート環境のルートスライスから、以前の稼働ブート環境をアクティブにします。

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

```
Do you want to fallback to activate boot environment c0t4d0s0
(yes or no)? yes
```

- 10 ディレクトリをアンマウントします。

```
# umount device_name
```

- 11 リブートします。

```
# init 6
```

前の稼働ブート環境がアクティブブート環境になります。

Live Upgrade ブート環境の管理 (タスク)

この章では、ブート環境のファイルシステムを最新の状態に維持したり、ブート環境を削除するなど、さまざまな管理タスクについて説明します。

注 - この章では、Oracle Solaris の機能である Live Upgrade の UFS ファイルシステムでの使用について説明します。管理作業の内容は、ZFS ブート環境でも同じです。UFS ファイルシステムを ZFS ルートプールに移行する、または ZFS ルートプールを作成およびインストールする手順については、[第 12 章「ZFS ルートプールのブート環境の作成」](#)を参照してください。

この章には次のセクションが含まれています。

- 132 ページの「Live Upgrade の管理の概要」
- 132 ページの「すべてのブート環境のステータスの表示」
- 133 ページの「以前に構成されたブート環境の更新」
- 134 ページの「スケジュールされた処理 (作成/アップグレード/コピー) の取り消し」
- 135 ページの「ブート環境の比較」
- 135 ページの「非アクティブブート環境の削除」
- 136 ページの「アクティブブート環境の名前の表示」
- 137 ページの「ブート環境の名前の変更」
- 138 ページの「ブート環境名に関連付ける説明の作成または変更」
- 139 ページの「ブート環境の構成の表示」

Live Upgrade の管理の概要

表 7-1 考えられる Live Upgrade 管理タスクの概要

タスク	説明	参照先
ステータスを表示します。	<ul style="list-style-type: none">■ ブート環境の状態 (アクティブ、アクティブ化の処理中、アクティブになるようにスケジュールされている、比較処理中) を表示します。■ アクティブブート環境と非アクティブブート環境を比較します。■ アクティブブート環境の名前を表示します。■ ブート環境の構成を表示します。	<ul style="list-style-type: none">■ 132 ページの「すべてのブート環境のステータスの表示」■ 135 ページの「ブート環境の比較」■ 136 ページの「アクティブブート環境の名前の表示」■ 139 ページの「ブート環境の構成の表示」
非アクティブブート環境を更新します。	ファイルシステムの構成を変更することなく、アクティブブート環境からファイルシステムを再度コピーします。	133 ページの「以前に構成されたブート環境の更新」
ブート環境を変更します。 スケジュールされているジョブを取り消します。	<ul style="list-style-type: none">■ ブート環境を削除します。■ ブート環境の名前を変更します。■ ブート環境の名前に関連付ける説明を作成または変更します。■ スケジュールされた作成、更新、またはコピーのジョブを取り消します。	<ul style="list-style-type: none">■ 135 ページの「非アクティブブート環境の削除」■ 137 ページの「ブート環境の名前の変更」■ 138 ページの「ブート環境名に関連付ける説明の作成または変更」■ 134 ページの「スケジュールされた処理 (作成/アップグレード/コピー) の取り消し」

ブート環境のための管理作業

すべてのブート環境のステータスの表示

ブート環境についての情報を表示するには、`lustatus` コマンドを使用します。ブート環境を指定しない場合は、システム上のすべてのブート環境のステータス情報が表示されます。

各ブート環境について、次の詳細情報が表示されます。

- Name - 各ブート環境の名前。
- Complete - コピー処理、作成処理とも進行中ではなく、ブート環境をブートできる状態であることを示します。作成処理またはアップグレード処理が進行中であったり失敗した場合などは、ステータスは未完了として示されます。たとえば、あるブート環境のコピー処理が進行中であるか、コピー処理がスケジュールされている場合は、そのブート環境は未完了とみなされます。
- Active - アクティブブート環境であるかどうかを示します。
- ActiveOnReboot - システムの次のリブート時にそのブート環境がアクティブになるかどうかを示します。
- CopyStatus - ブート環境の作成またはコピーがスケジュールされているか、アクティブか、またはアップグレード処理中かを示します。ステータスが SCHEDULED の場合、Live Upgrade のコピー、名前変更、アップグレードの各処理を行うことはできません。

この例では、すべてのブート環境のステータスが表示されます。特定のブート環境のステータスを表示するには、ブート環境の名前を指定してコマンドを発行します。

```
# su
# 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	-

サンプル出力によると、disk_b_S9a は未完了であり、disk_b_S10database は Live Upgrade による処理中であるため、これらに対してコピー、名前変更、アップグレードの処理を行うことはできません。

以前に構成されたブート環境の更新

「Copy」メニューまたは lumake コマンドを使用して、以前に構成されたブート環境の内容を更新できます。アクティブ(ソース)ブート環境のファイルシステムがターゲットブート環境にコピーされます。ターゲット上にあったデータは破棄されます。コピー元のブート環境のステータスは、Complete である必要があります。ブート環境のステータスを確認する方法の詳細については、[132 ページの「すべてのブート環境のステータスの表示」](#)を参照してください。

コピー作業はあとで行われるようにスケジュールできます。スケジュールできるのは一度に 1 つのジョブだけです。スケジュールされたコピーを取り消す方法については、[134 ページの「スケジュールされた処理 \(作成/アップグレード/コピー\) の取り消し」](#)を参照してください。

lumake コマンドの構文は次のとおりです。

```
# lumake -n BE-name [-s source-BE] [-t time] [-m email-address]
```

-n BE-name ファイルシステムを書き換えるブート環境の名前を指定します。

-s source-BE (オプション) ターゲットブート環境にコピーするファイルシステムがあるソースブート環境の名前を指定します。このオプションを省略すると、lumake は現在のブート環境をソースとして使用します。

-t time (オプション) 指定されたブート環境で指定された時間にファイルシステムのコピーを行うバッチジョブを設定します。時間引数の書式を設定する方法については、[at\(1\)](#) のマニュアルページを参照してください。

-m email-address (オプション) コマンドが完了した時点で、指定したアドレスに lumake の出力を電子メールで送信できます。email-address はチェックされません。このオプションは、-t と併用する必要があります。

例 7-1 以前に構成されたブート環境の更新

この例では、first_disk のファイルシステムが second_disk にコピーされます。処理が完了した時点で、電子メールが Joe@anywhere.com 宛に送信されます。

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

first_disk 上のファイルが second_disk にコピーされ、通知の電子メールが送信されます。スケジュールされたコピーを取り消す方法については、[134 ページの「スケジュールされた処理 \(作成/アップグレード/コピー\) の取り消し」](#)を参照してください。

スケジュールされた処理 (作成/アップグレード/コピー) の取り消し

ブート環境のスケジュールされた作成、アップグレード、またはコピーのジョブは、そのジョブの開始前に取り消すことができます。lumake コマンドを使用すると、ジョブをスケジュールできます。システムでスケジュールできるジョブは一度に 1 つだけです。

スケジュールされたジョブを取り消すには、スーパーユーザーまたは同等の役割になり、lucancel コマンドを実行します。

ブート環境の比較

アクティブブート環境とその他のブート環境の相違を確認するには、`lucompare` コマンドを使用します。比較を行うには、非アクティブブート環境が **Complete** 状態である必要があります、コピージョブが保留中であってはけません。[132 ページの「すべてのブート環境のステータスの表示」](#)を参照してください。

`lucompare` コマンドは、非大域ゾーンの内容が含まれているブート環境の比較を行います。

`lumount` または `mount` を使用してマウントされたパーティションのあるブート環境は、指定できません。

`lucompare` コマンドの構文は次のとおりです。

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

-i infile *infile* 中に指定されたファイルを比較します。比較するファイルは、絶対パスで指定する必要があります。ファイルのエントリがディレクトリである場合、比較はディレクトリに対して再帰的に行われます。このオプションまたは **-t** のいずれか一方を使用できます (両方は使用できません)。

-t バイナリ以外のファイルだけを比較します。この比較では、ファイルごとに `file(1)` コマンドを使用してそのファイルがテキストファイルであるかどうかを確認します。ユーザーは、このオプションまたは **-i** のいずれか一方を使用できます (両方は使用できません)。

-o outfile 相違についての出力を *outfile* にリダイレクトします。

BE-name アクティブブート環境と比較するブート環境の名前を指定します。

例 7-2 ブート環境の比較

この例では、`first_disk` ブート環境 (ソース) と `second_disk` ブート環境が比較され、結果がファイルに出力されます。

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

非アクティブブート環境の削除

ブート環境を削除するには、`ludelete` コマンドを使用します。次の制限に注意してください。

- アクティブブート環境および次のリブートでアクティブになるブート環境は、削除できません。

- 削除するブート環境は完了状態でなければなりません。完了ステータスのブート環境とは、ステータスを変更する処理が終了している環境を指します。
[132 ページの「すべてのブート環境のステータスの表示」](#)に、ブート環境のステータスを確認する方法についての説明があります。
- `lumount` を使用してファイルシステムをマウントしているブート環境を削除することはできません。
- x86 のみ: **Solaris 10 1/06** 以降のリリースでは、アクティブな GRUB メニューがあるブート環境を削除することはできません。ブート環境を再使用するには、`lumake` コマンドまたは `luupgrade` コマンドを使用します。どのブート環境にアクティブな GRUB メニューがあるかを調べるには、『[Oracle Solaris の管理: 基本管理](#)』の第 13 章『[Oracle Solaris ブートアーカイブの管理 \(タスク\)](#)』を参照してください。

`ludelete` コマンドの構文は次のとおりです。

```
# ludelete BE-name
```

BE-name 削除する非アクティブブート環境の名前を指定します。

例 7-3 非アクティブブート環境の削除

この例では、ブート環境 `second_disk` が削除されます。

```
# ludelete second_disk
```

アクティブブート環境の名前の表示

現在稼動しているブート環境の名前を表示するには、`lucurr` コマンドを使用します。システム上に構成されたブート環境がない場合は、「No Boot Environments are defined」というメッセージが表示されます。`lucurr` で表示されるのは現在のブート環境の名前だけです。次のリブート時にアクティブになるブート環境の名前は表示されません。ブート環境のステータスを確認する方法については、[132 ページの「すべてのブート環境のステータスの表示」](#)を参照してください。

例 7-4 アクティブブート環境の名前の表示

この例では、現在のブート環境の名前が表示されます。

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


ブート環境の名前の変更

ブート環境の名前の変更は、ブート環境の Oracle Solaris リリースを別のリリースにアップグレードする場合に役立つことがあります。たとえば、オペレーティングシステムのアップグレード処理後に、ブート環境の名前を `solaris8` から `solaris10` に変更できます。

非アクティブブート環境の名前を変更するには、`lurename` コマンドを使用します。

x86 のみ - Solaris 10 1/06 以降のリリースでは、GRUB メニューは、「Rename」メニューまたは `lurename` コマンドを使用すると自動的に更新されます。更新された GRUB メニューでは、ブートエントリのリストにブート環境の名前が表示されます。GRUB メニューの詳細は、[57 ページの「複数のブート環境のブート」](#)を参照してください。

GRUB メニューの `menu.lst` ファイルの場所を判別する方法については、『[Oracle Solaris の管理: 基本管理](#)』の第 13 章「[Oracle Solaris ブートアーカイブの管理 \(タスク\)](#)」を参照してください。

ブート環境の命名に関する次の制限事項に注意してください。

- 名前の長さは、30 文字以内にする必要があります。
- 名前は、英数字またはほかの ASCII 文字 (UNIX シェルで特別な意味を持つ文字を除く) だけで構成できます。詳細については、`sh(1)` のマニュアルページの「クォート」セクションを参照してください。
- 名前に使用できるのは、8 ビットで表現できるシングルバイトの文字だけです。
- 名前は、システム上で一意となるように指定する必要があります。
- ブート環境の名前を変更するためには、そのステータスが `Complete` である必要があります。ブート環境のステータスを確認する方法の詳細については、[132 ページの「すべてのブート環境のステータスの表示」](#)を参照してください。
- `lumount` または `mount` を使用してファイルシステムをマウントしているブート環境の名前は変更できません。

`lurename` コマンドの構文は次のとおりです。

```
# lurename -e BE-name -n new-name
```

`-e BE-name` 変更する非アクティブブート環境の名前を指定します。

`-n new-name` 非アクティブブート環境の新しい名前を指定します。

この例では、`second_disk` が `third_disk` に変更されます。

```
# lurename -e second_disk -n third_disk
```

ブート環境名に関連付ける説明の作成または変更

ブート環境名に説明を付けることができます。この説明によって名前が置き換わることはありません。ブート環境名は長さや文字に制限がありますが、この説明は長さ、内容に制限がありません。シンプルなテキストでも、gif ファイルのような複雑なものでもかまいません。この説明は、次の時点で作成できます。

- `lucreate` コマンドと `-A` オプションを使用してブート環境を作成するとき。詳細は、[67 ページの「ブート環境をはじめて作成する方法」](#)を参照してください。
- `ludesc` コマンドを使用してブート環境を作成したあと。詳細は、[ludesc\(1M\)](#) のマニュアルページを参照してください。

`ludesc` コマンドの構文は次のとおりです。

```
# /usr/sbin/ludesc -n BE-name 'BE-description' -f filename
-n BE-name          ブート環境名を指定します。
BE-description      名前に関連付ける新しい説明を指定します。
filename            ブート環境名に関連付けるファイルを指定します。
```

例 7-5 テキストを使用してブート環境名に説明を加える

この例では、`second_disk` というブート環境に説明が加えられています。この説明は、単一引用符で囲まれたテキストで記述されます。

```
# /usr/sbin/ludesc -n second_disk 'Oracle Solaris 10 8/11 test build'
```

例 7-6 ファイルを使用してブート環境名に説明を加える

この例では、`second_disk` というブート環境に説明が加えられています。説明は、gif ファイル内に入っています。

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

例 7-7 説明からブート環境名を表示する

この例では、説明を指定した `-A` オプションを使用することでブート環境名 `second_disk` を表示しています。

```
# /usr/sbin/ludesc -A 'Oracle Solaris 10 8/11 test build'
second_disk
```

例7-8 ファイル内の説明からブート環境名を表示する

この例では、`-f` オプションと、説明を含むファイルの名前を使用することでブート環境の名前 `second_disk` を表示しています。

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

例7-9 名前からブート環境の説明を表示する

この例では、ブート環境名を指定した `-n` オプションを使用することで説明を表示しています。

```
# /usr/sbin/ludesc -n second_disk
Oracle Solaris 10 8/11 test build
```

ブート環境の構成の表示

ブート環境の構成を表示するには、`lufslist` コマンドを使用します。出力される情報は、ブート環境マウントポイントごとの、ディスクスライス(ファイルシステム)、ファイルシステムの種類、およびファイルシステムのサイズです。

`lufslist` コマンドの構文は次のとおりです。

```
# lufslist -n BE-name
```

BE-name ファイルシステムの詳細を表示するブート環境の名前を指定します。

次の例ではリストを表示しています。

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

注- 非大域ゾーンが含まれているリストの例については、[156 ページの「ブート環境の非大域ゾーンのファイルシステムの構成を表示する」](#)を参照してください。

非大域ゾーンがインストールされているシステムにおける Oracle Solaris OS のアップグレード

この章では、Live Upgrade を使用して、非大域ゾーンがインストールされているシステムをアップグレードする方法について説明します。

注- この章では、UFS ファイルシステム用の Live Upgrade について説明します。非大域ゾーンがある UFS ファイルシステムを ZFS ルートプールに移行する手順については、[第 13 章「非大域ゾーンがインストールされている ZFS での Live Upgrade」](#)を参照してください。

この章には次のセクションが含まれています。

- Live Upgrade を使用して非大域ゾーンが含まれているシステムをアップグレードする場合の変更のサマリーについては、[142 ページの「Live Upgrade によるアップグレードおよびインストール済みの非大域ゾーン \(概要\)」](#)を参照してください。
- Live Upgrade を使用して新しいブート環境を作成し、システムをアップグレードする手順については、[149 ページの「非大域ゾーンがインストールされている場合のブート環境の作成とアップグレード \(タスク\)」](#)を参照してください。
- Live Upgrade を使用して新しいブート環境を作成し、システムをアップグレードする例と簡単な説明については、[154 ページの「非大域ゾーンがインストールされているシステムのアップグレード \(例\)」](#)を参照してください。
- 非大域ゾーンの作成の概要については、『[Oracle Solaris の管理: Oracle Solaris コンテナ-リソース管理と Oracle Solaris ゾーン](#)』を参照してください。

Live Upgrade によるアップグレードおよびインストール済みの非大域ゾーン (概要)

Solaris 10 8/07 リリース以降では、Live Upgrade を使用して、非大域ゾーンが含まれているシステムのアップグレードまたはパッチ適用を行うことができます。システムに非大域ゾーンが含まれている場合は、アップグレードやパッチを追加するプログラムとして、Live Upgrade を推奨します。ほかのアップグレードプログラムでは、膨大なアップグレード時間が必要となる場合があります。これは、アップグレードの実行に要する時間が、インストールされている非大域ゾーンの数に比例して増加するからです。Live Upgrade を使ってシステムにパッチを適用する場合は、システムをシングルユーザーモードにする必要がないため、システムの稼働時間を最大化できます。次のリストは、非大域ゾーンがインストールされているシステムに対応するための変更をまとめたものです。

- 新しいパッケージ SUNWlucfg をほかの Live Upgrade パッケージ SUNWlur および SUNWluu とともにインストールする必要があります。このパッケージは、非大域ゾーンがインストールされているシステムだけでなく、どのシステムにも必要です。
- 現在稼働しているブート環境から新しいブート環境を作成する方法は以前のリリースの場合と同じですが、例外が1つあります。非大域ゾーン内の共有ファイルシステムに対して宛先ディスクスライスを指定できます。詳細は、[149 ページの「非大域ゾーンがインストールされている場合のブート環境の作成とアップグレード \(タスク\)」](#)を参照してください。
- lumount コマンドは、非大域ゾーンが、非アクティブブート環境に存在する、それらに対応するファイルシステムにアクセスできるようにします。大域ゾーン管理者が lumount コマンドを使って非アクティブブート環境をマウントすると、同様にブート環境が非大域ゾーン用にマウントされます。[158 ページの「非大域ゾーンが含まれるシステムで lumount コマンドを使用する」](#)を参照してください。
- ブート環境の比較機能が向上しました。lucompare コマンドは、非大域ゾーンの内容が含まれているブート環境の比較を行うようになりました。[157 ページの「非大域ゾーンがインストールされているシステムのブート環境を比較する」](#)を参照してください。
- lufslist コマンドによるファイルシステムの表示機能が向上し、大域ゾーンと非大域ゾーンの両方のファイルシステムの一覧が表示されるようになりました。[156 ページの「ブート環境の非大域ゾーンのファイルシステムの構成を表示する」](#)を参照してください。

Oracle Solaris ゾーンと Live Upgrade について

Oracle Solaris ゾーンパーティション分割テクノロジーは、オペレーティングシステムサービスを仮想化し、実行中のアプリケーションのために隔離されたセキュアな環

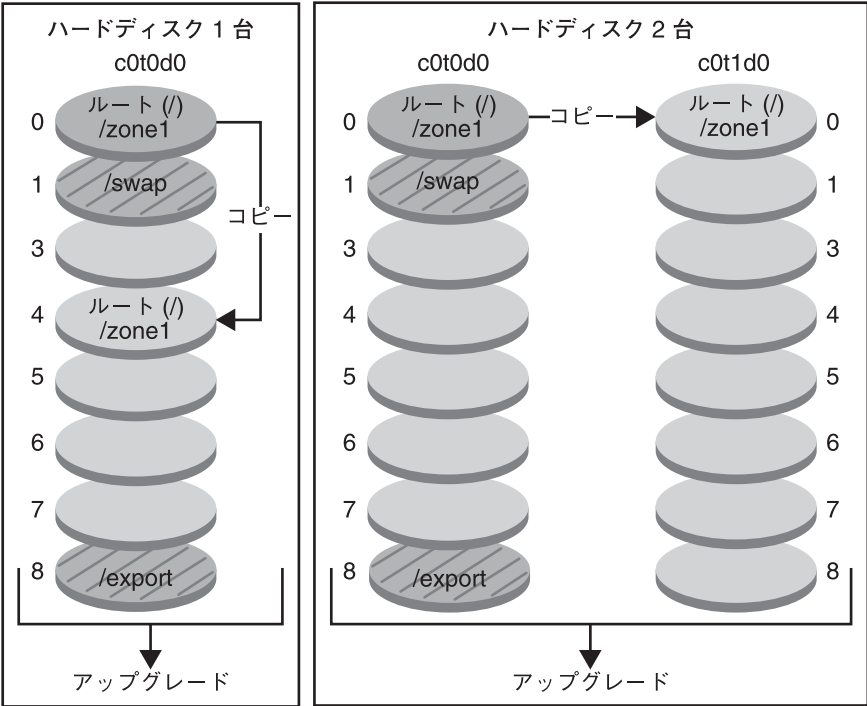
境を提供するために使用されます。非大域ゾーンは、Oracle Solaris OS の 1 つのインスタンス、つまり大域ゾーン内で作成される仮想化されたオペレーティングシステム環境です。非大域ゾーンを作成すると、アプリケーション実行環境が生成されます。このアプリケーション実行環境内のプロセスは、システムのほかの部分から隔離されます。

Live Upgrade は、現在稼働しているシステムを新しいスライス上にコピーするためのメカニズムです。非大域ゾーンがインストールされている場合は、大域ゾーンのファイルシステムとともにその非大域ゾーンを非アクティブブート環境にコピーできます。

次の図に、大域ゾーンのファイルシステムとともに非アクティブブート環境にコピーされる非大域ゾーンを示します。

図 8-1 ブート環境の作成 – 非大域ゾーンのコピー

ブート環境の作成 –
非大域ゾーンのコピー



ディスク 1 台の場合のコマンド:
lucreate -c bootenv1 \
-m /dev/dsk/c0t0d0s4:ufs \
-n bootenv2

ディスク 2 台の場合のコマンド:
lucreate -c bootenv1 \
-m /dev/dsk/c0t1d0s0:ufs \
-n bootenv2

- 現在のリリース X
クリティカルなファイルシステムのルート (/)
- 非アクティブなリリース X
クリティカルなファイルシステムのルート (/)
- 共有ファイルシステム

- ディスクが 1 つだけのシステムの例では、ルート (/) ファイルシステムが c0t0d0s4 にコピーされます。このファイルシステムに関連付けられている非大域ゾーンもすべて s4 にコピーされます。/export ファイルシステムおよび /swap ボリュームは、現在のブート環境 bootenv1 と非アクティブブート環境 bootenv2 との間で共有されます。lucreate コマンドは次のとおりです。

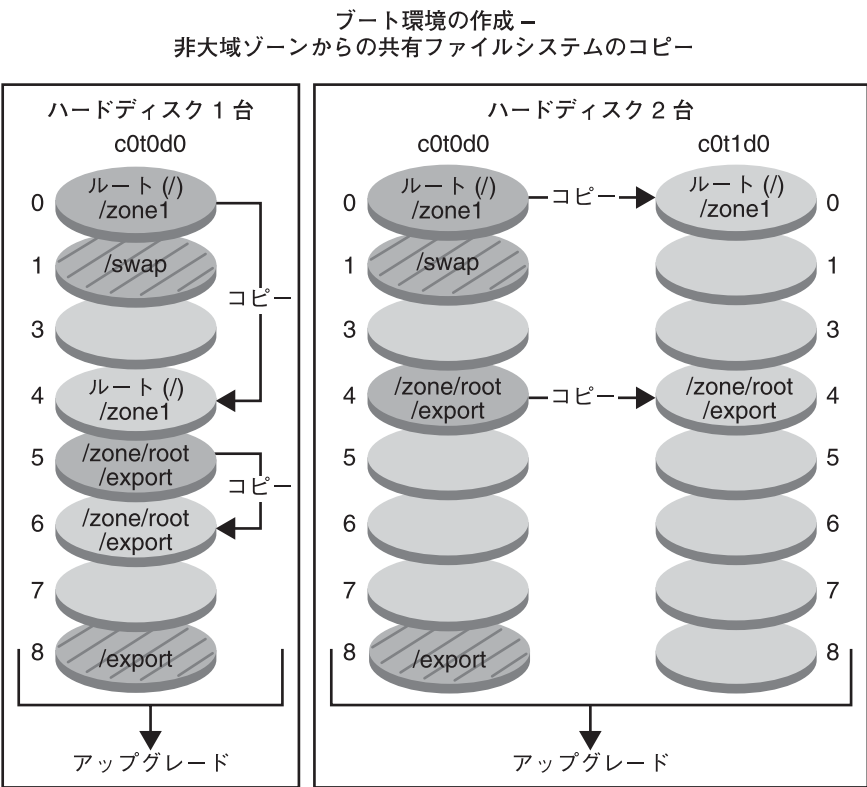
lucreate -c bootenv1 -m /dev/dsk/c0t0d0s4:ufs -n bootenv2

- ディスクが2つあるシステムの例では、ルート (/) ファイルシステムが `c0t1d0s0` にコピーされます。このファイルシステムに関連付けられている非大域ゾーンもすべて `s0` にコピーされます。`/export` ファイルシステムおよび `/swap` ボリュームは、現在のブート環境 `bootenv1` と非アクティブブート環境 `bootenv2` との間で共有されます。`lucreate` コマンドは次のとおりです。

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

次の図に、非大域ゾーンが非アクティブブート環境にコピーされる様子を示します。

図 8-2 ブート環境の作成 – 非大域ゾーンからの共有ファイルシステムのコピー



ディスク 1 台の場合のコマンド:

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

ディスク 2 台の場合のコマンド:

```
# lucreate -c bootenv1 \  
-m /dev/dsk/c0t1d0s1:ufs \  
-m /export:/dev/dsk/c0t1d0s4:ufs:zone1  
-n bootenv2
```

- ☒ 現在のリリース X
クリティカルなファイルシステムのルート (/)
- ☐ 非アクティブなリリース X
クリティカルなファイルシステムのルート (/)
- ☒ 共有ファイルシステム

- ディスクが1つだけのシステムの例では、ルート (/) ファイルシステムが c0t0d0s4 にコピーされます。このファイルシステムに関連付けられている非大域ゾーンもすべて s4 にコピーされます。非大域ゾーン zone1 には、zonecfg add fs コマンドで作成された独立したファイルシステムがあります。ゾーンパスは /zone1/root/export です。このファイルシステムは、非アクティブブート環境で共有されないように、個別のスライス c0t0d0s6 上に配置されます。/export

ファイルシステムおよび /swap ボリュームは、現在のブート環境 `bootenv1` と非アクティブブート環境 `bootenv2` との間で共有されます。lucreate コマンドは次のとおりです。

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

- ディスクが2つあるシステムの例では、ルート (/) ファイルシステムが `c0t1d0s0` にコピーされます。このファイルシステムに関連付けられている非大域ゾーンもすべて `s0` にコピーされます。非大域ゾーン `zone1` には、`zonecfg add fs` コマンドで作成された独立したファイルシステムがあります。ゾーンパスは `/zone1/root/export` です。このファイルシステムは、非アクティブブート環境で共有されないように、個別のスライス `c0t1d0s4` 上に配置されます。/export ファイルシステムおよび /swap ボリュームは、現在のブート環境 `bootenv1` と非アクティブブート環境 `bootenv2` との間で共有されます。lucreate コマンドは次のとおりです。

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

非大域ゾーンで **Live Upgrade** を使用するためのガイドライン (計画)

ゾーンがインストールされているシステムで Live Upgrade を使用する場合は、次の問題を考慮してください。lucreate および lumount 操作の実行中にゾーン状態が遷移しないようにすることが非常に重要です。

- ある特定の非大域ゾーンが実行されていないときに、lucreate コマンドを使用して非アクティブブート環境を作成した場合、そのゾーンは lucreate 操作が完了するまでブートできません。
- ある特定の非大域ゾーンが実行されているときに、lucreate コマンドを使用して非アクティブブート環境を作成した場合は、lucreate 操作が完了するまで、そのゾーンを停止またはリブートしないでください。
- lumount コマンドを使用して非アクティブブート環境をマウントした場合、その lumount 操作より前に実行されていたゾーンは実行を継続できますが、非大域ゾーンをブートしたり、リブートすることはできません。
- 非大域ゾーンは、非大域ゾーン管理者だけでなく大域ゾーン管理者にも制御できるため、相互に干渉することを避けるため、lucreate または lumount 操作の実行中には、すべてのゾーンを停止してください。

大域ゾーン管理者が、Live Upgrade を使用したアップグレードについて非大域ゾーン管理者に通知しないと、問題が発生する可能性があります。

Live Upgrade 操作の進行中は、非大域ゾーン管理者が介入してはいけません。アップグレードは、アップグレードによって発生する変更に対処する予定の管理者の作業

に影響を及ぼします。ゾーン管理者は、すべてのローカルパッケージが一連の操作を通じて確実に安定しているようにし、構成ファイルの調整といったアップグレード後のタスクをすべて行い、通常はシステムの機能停止を避けたスケジュールを立てる必要があります。

たとえば、大域ゾーン管理者が `lucreate` コマンドを使用してファイルシステムをコピーしているときに、非大域ゾーン管理者がパッケージを追加すると、その新しいパッケージはファイルシステムとともにコピーされず、非大域ゾーン管理者は問題の発生に気づきません。

非大域ゾーンが独立したファイルシステム上にある場合にブート環境を作成する

現在稼動しているブート環境から新しいブート環境を作成する方法は、1つの例外を除いて、以前のリリースの場合と同じです。非大域ゾーン内の共有ファイルシステムに対して宛先ディスクスライスを指定できます。この例外は、次の条件のもとで発生します。

- 現在のブート環境で `zonecfg add fs` コマンドが使用され、非大域ゾーンに対して独立したファイルシステムが作成された場合
- この個別のファイルシステムが、`/zone/root/export` などの共有ファイルシステム上にある場合

この独立したファイルシステムが新しいブート環境で共有されないようにするため、`lucreate` コマンドでは、非大域ゾーンの独立したファイルシステムに対して宛先スライスを指定できます。`-m` オプションの引数には、新しいオプションフィールド `zonename` が追加されました。この新しいフィールドは、非大域ゾーンの個別のファイルシステムを新しいブート環境の個々のスライス上に配置します。非大域ゾーンに独立したファイルシステムを設定する方法の詳細は、[zonecfg\(1M\)](#) のマニュアルページを参照してください。

注-デフォルトでは、クリティカルファイルシステム(ルート(/)、`/usr`、`/opt` ファイルシステム)以外のすべてのファイルシステムが、現在のブート環境と新しいブート環境との間で共有されます。このため、アクティブブート環境内の共有ファイルを更新すると、非アクティブブート環境のデータも更新されます。たとえば、`/export` ファイルシステムは共有ファイルシステムです。`-m` オプションと `zonename` オプションを使用すると、非大域ゾーンのファイルシステムが別のスライスにコピーされ、データは共有されません。このオプションを使用すると、`zonecfg add fs` コマンドを使って作成した非大域ゾーンのファイルシステムがブート環境間で共有されなくなります。

非大域ゾーンがインストールされている場合のブート環境の作成とアップグレード(タスク)

次のセクションでは、非大域ゾーンがインストールされている場合の段階的なアップグレード手順について説明します。

- [149 ページの「非大域ゾーンがシステムにインストールされている場合の Live Upgrade でのアップグレード\(タスク\)」](#)

例と簡潔な手順については、[154 ページの「非大域ゾーンがインストールされているシステムのアップグレード\(例\)」](#)を参照してください。

▼ 非大域ゾーンがシステムにインストールされている場合の **Live Upgrade** でのアップグレード(タスク)

次の手順では、非大域ゾーンがインストールされているシステムを Live Upgrade でアップグレードするための詳しい指示を提供します。

- 1 **Live Upgrade** をはじめて実行する前に、インストールメディアから最新の **Live Upgrade** パッケージをインストールし、**My Oracle Support** ナレッジキュメント **1004881.1 – Live Upgrade Software Patch Requirements** (以前の **206844**) に記載されているパッチをインストールする必要があります。[My Oracle Support Web](#) サイトでナレッジキュメント **1004881.1 – Live Upgrade Software Patch Requirements** (以前の **206844**) を検索してください。

最新のパッケージとパッチにより、リリースに最新のバグ修正と新機能がすべて含まれるようになります。新しいブート環境の作成に進む前に、システムに関連するすべてのパッチを必ずインストールしてください。

次のサブステップは、**My Oracle Support** ナレッジキュメント **1004881.1 – Live Upgrade Software Patch Requirements** (以前の **206844**) の手順を説明したものです。

- a. スーパーユーザーになるか、同等の役割を引き受けます。

注-役割には、認証と特権コマンドが含まれます。役割の詳細については、『[Solaris のシステム管理: セキュリティーサービス](#)』の「[RBAC の構成\(タスクマップ\)](#)」を参照してください。

- b. **My Oracle Support Web** サイトで、ナレッジキュメント **1004881.1 – Live Upgrade Software Patch Requirements** (以前の **206844**) の手順に従って、**Live Upgrade** パッケージを削除して追加します。

次の手順は、パッケージの削除と追加についてのナレッジドキュメントの手順をまとめたものです。

- 既存の Live Upgrade パッケージを削除します。

3つの Live Upgrade パッケージ SUNWluu、SUNWlur、および SUNWlucfg は、Live Upgrade を使ってアップグレードまたはパッチを適用するのに必要なソフトウェアを構成します。これらのパッケージには、既存のソフトウェア、新しい機能、およびバグ修正が含まれています。Live Upgrade を使用する前に、既存パッケージを削除しないで新しいパッケージをシステムにインストールすると、ターゲットリリースへのアップグレードまたはパッチ適用は失敗します。SUNWlucfg パッケージは、**Solaris 10 8/07** 以降のリリースの新機能です。Solaris 10 8/07 より前のリリースから Live Upgrade パッケージを使用する場合は、このパッケージを削除する必要はありません。

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- 新しい Live Upgrade パッケージをインストールします。

パッケージをインストールするには、インストール DVD または CD にある liveupgrade20 コマンドを使用します。liveupgrade20 コマンドを使用する場合は、Java ソフトウェアが必要です。システムに Java ソフトウェアがインストールされていない場合は、pkgadd コマンドを使用してパッケージをインストールする必要があります。詳細は、My Oracle Support ナレッジドキュメントを参照してください。

- Oracle Solaris Operating System DVD を使用している場合は、ディレクトリを変更してインストーラを実行します。
 - ディレクトリを変更します。

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

注 - SPARC ベースのシステムの場合、Solaris 10 10/08 リリースより前のリリースではインストーラへのパスが異なります。

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

- インストーラを実行します。

```
# ./liveupgrade20
```

Oracle Solaris インストールプログラムの GUI が表示されます。スクリプトを使用する場合は、-noconsole オプションと -nodisplay オプションを使用することで、GUI が表示されるのを防ぐことができます。

- Oracle Solaris Software - 2 CD を使用する場合は、パスを変更せずにインストーラを実行できます。

```
% ./installer
```

- パッケージが正常にインストールされていることを確認します。

```
# pkgchk -v SUNWlucfg SUNWlur SUNWluu
```

- c. ローカルディスクにパッチを保存する場合は、`/var/tmp/lupatches`などのディレクトリを作成します。

- d. **My Oracle Support**のWeb サイトから、パッチの一覧を取得します。

- e. この例のようにパッチディレクトリに移動します。

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

- f. パッチをインストールします。

```
# patchadd -M path-to-patchespatch-id patch-id
```

`path-to-patches`は、`/var/tmp/lupatches`などのパッチディレクトリへのパスです。`patch_id`はパッチの番号です。複数のパッチ名を指定する場合は、スペースで区切ります。

注 - パッチはナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) に指定されている順序で適用する必要があります。

- g. 必要に応じてシステムをリブートします。いくつかのパッチは、有効にするためにリブートする必要があります。

x86のみ: システムをリブートする必要があります。それ以外の場合、Live Upgradeが失敗します。

```
# init 6
```

以上で新しいブート環境を正しく作成するために必要なパッケージとパッチがインストールされました。

- 2 新しいブート環境を作成します。

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

-n *BE_name*

作成するブート環境の名前。*BE_name*は、システム上で一意となるように指定する必要があります。

-A '*BE_description*'

(オプション) ブート環境名 (*BE_name*) の説明を記述できます。記述の長さ、使用できる文字に制限はありません。

-c *BE_name*

アクティブブート環境に名前 *BE_name* を割り当てます。このオプションは省略可能で、最初のブート環境を作成する場合だけ使用されます。`lucreate`を初めて実行する場合に `-c` オプションを省略すると、デフォルトの名前が作成されます。

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

新しいブート環境のファイルシステム構成を `vfstab` で指定します。-m に引数として指定されるファイルシステムは、同じディスク上のファイルシステムでも、複数のディスク上のファイルシステムでも構いません。このオプションは、作成するファイルシステムの数だけ使用します。

- `mountpoint` には、任意の有効なマウントポイント、またはスワップパーティションを示す - (ハイフン) を指定できます。
- `device` フィールドには、次のいずれかを指定できます。
 - ディスクデバイスの名前。/dev/dsk/cwtxdysz の形式で表されます
 - Solaris ポリウムマネージャーのポリウム名。/dev/md/dsk/dnum の形式で表されます
 - Veritas Volume Manager のポリウム名。/dev/md/vxfs/dsk/dnum の形式で表されます
 - キーワード `merged`。指定されたマウントポイントのファイルシステムがその親とマージされることを示します
- `fs_options` フィールドには、次のいずれかを指定できます。
 - `ufs`: UFS ファイルシステムを示します。
 - `vxfs`: Veritas ファイルシステムを示します。
 - `swap`: スワップポリウムを示します。スワップマウントポイントはハイフン (-) で表します。
 - 論理デバイス (ミラー) であるファイルシステムについては、いくつかのキーワードを使って、そのファイルシステムに対して実行するアクションを指定できます。論理デバイスの作成、論理デバイスの構成変更、論理デバイスの削除などを行うキーワードがあります。これらのキーワードの説明は、[82 ページの「RAID-1 ポリウム \(ミラー\) を持つブート環境を作成する方法」](#)を参照してください。
 - `zonename` は、非大域ゾーンの独立したファイルシステムが個別のスライス上に配置されるように指定します。このオプションは、そのゾーンの独立したファイルシステムが /zone1/root/export などの共有ファイルシステム内にある場合に使用します。このオプションは、そのゾーンの独立したファイルシステムを新しいスライスにコピーして、このファイルシステムが共有されないようにします。独立したファイルシステムは、`zonecfg add fs` コマンドを使って作成したものです。

次の例では、`newbe` という新しいブート環境が作成されます。ルート (/) ファイルシステムは、`c0t1d0s4` 上に配置されます。現在のブート環境にある非大域ゾーンはすべて、新しいブート環境にコピーされます。`zone1` という非大域ゾーンに対して個別のマウントポイントが `c0t1d0s1` 上に指定されます。

注-デフォルトでは、クリティカルファイルシステム(ルート(/)、/usr、/opt ファイルシステム)以外のすべてのファイルシステムが、現在のブート環境と新しいブート環境との間で共有されます。/export ファイルシステムは共有ファイルシステムです。-m オプションを使用すると、非大域ゾーンのファイルシステムが別のスライス上に配置され、データは共有されません。このオプションは、zonecfg add fs コマンドを使って作成したゾーンのファイルシステムがブート環境間で共有されないようにします。詳細は、[zonecfg\(1M\)](#)のマニュアルページを参照してください。

```
# lucreate -n newbe -m /:/dev/dsk/c0t1d0s4:ufs -m /export:/dev/dsk/c0t1d0s1:ufs:zone1
```

3 ブート環境をアップグレードします。

アップグレードに使用するオペレーティングシステムイメージはネットワークから取得します。

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

-u ブート環境のオペレーティングシステムイメージをアップグレードします。

-n BE_name アップグレード対象のブート環境の名前を指定します。

-s os_image_path オペレーティングシステムイメージが置かれているディレクトリのパス名を指定します。

この例では、新しいブート環境 newbe がネットワークインストールイメージからアップグレードされます。

```
# luupgrade -n newbe -u -s /net/server/export/Solaris_10/combined.solaris_wos
```

4 (オプション)ブート環境がブート可能であることを確認します。

lustatus コマンドは、ブート環境の作成が完了してブート可能であるかどうかを報告します。

```
# lustatus
```

boot environment Name	Is Complete	Active Now	Active OnReboot	Can Delete	Copy Status
c0t1d0s0	yes	yes	yes	no	-
newbe	yes	no	no	yes	-

5 新しいブート環境をアクティブにします。

```
# luactivate BE_name
```

BE_name は、アクティブにするブート環境の名前を示します。

注-x86 ベースのシステムの場合、ブート環境をはじめてブートするときには `luactivate` コマンドを使用する必要があります。これ以降は、Oracle Solaris の機能である GRUB メニューからブート環境を選択することでアクティブにできます。手順については、[117 ページの「x86: GRUB メニューを使ったブート環境のアクティブ化」](#)を参照してください。

ブート環境を正常にアクティブにするためには、そのブート環境がいくつかの条件を満たしている必要があります。詳細は、[113 ページの「ブート環境のアクティブ化」](#)を参照してください。

6 リブートします。

```
# init 6
```



注意-リブートには `init` か `shutdown` コマンドを使用してください。 `reboot` や `halt`、`uadmin` コマンドを使用すると、ブート環境の切り替えは行われません。最後にアクティブになっていたブート環境が再びブートされます。

ブート環境が切り替わり、新しいブート環境が現在のブート環境になります。

7 (オプション)別のブート環境に戻します。

新しいブート環境が実行可能でない場合、または別のブート環境に切り替える必要がある場合は、[第6章「障害回復:元のブート環境へのフォールバック\(タスク\)」](#)を参照してください。

非大域ゾーンがインストールされているシステムのアップグレード (例)

次の手順では、Live Upgrade を使ってアップグレードする例と簡単な説明を示します。

手順の詳しい説明については、[149 ページの「非大域ゾーンがシステムにインストールされている場合の Live Upgrade でのアップグレード\(タスク\)」](#)を参照してください。

非大域ゾーンがシステムにインストールされている場合の **Live Upgrade** によるアップグレード

次の例では、非大域ゾーンがインストールされているシステムをアップグレードする手順について簡潔に説明します。この例では、Oracle Solaris 10 リリースが稼働し

ているシステムで、`lucreate` コマンドを使って新しいブート環境を作成します。このシステムには非大域ゾーンがインストールされており、そのゾーンの独立したファイルシステムは共有ファイルシステム `zone1/root/export` 上にあります。この新しいブート環境を、`luupgrade` コマンドを使用して Oracle Solaris 10 8/11 リリースにアップグレードします。アップグレードしたブート環境を、`luactivate` コマンドによってアクティブにします。

注- この手順では、システムでボリュームマネージャーを実行していると仮定しています。ボリュームマネージャーを使用したリムーバブルメディアの管理の詳細は、『[Oracle Solaris の管理: デバイスとファイルシステム](#)』を参照してください。

1. 必要なパッチをインストールします。

<http://support.oracle.com> (My Oracle Support) を調べて最新の更新済みパッチリストであることを確認してください。My Oracle Support で、ナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) を検索してください。この例では、`/net/server/export/patches` がパッチのパスです。

```
# patchadd /net/server/export/patches
# init 6
```

2. **Live Upgrade** パッケージを現在のブート環境から削除します。

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

3. **Oracle Solaris DVD** または **CD** を挿入します。次に、ターゲットリリースから交換用の **Live Upgrade** パッケージをインストールします。

```
# pkgadd -d /cdrom/cdrom0/Solaris_10/Product SUNWlucfg SUNWlur SUNWluu
```

4. ブート環境を作成します。

次の例では、`newbe` という新しいブート環境が作成されます。ルート (`/`) ファイルシステムは、`c0t1d0s4` 上に配置されます。現在のブート環境にある非大域ゾーンはすべて、新しいブート環境にコピーされます。`zonecfg add fs` コマンドを使って独立したファイルシステムを `zone1` に対して作成してあります。この独立したファイルシステム `/zone/root/export` は、個別のスライス `c0t1d0s1` 上に配置されます。このオプションを使用すると、独立したファイルシステムが現在のブート環境と新しいブート環境の間で共有されなくなります。

```
# lucreate -n newbe -m /:/dev/dsk/c0t1d0s4:ufs -m /export:/dev/dsk/c0t1d0s1:ufs:zone1
```

5. 新しいブート環境をアップグレードします。

この例では、`/net/server/export/Solaris_10/combined.solaris_wos` がネットワークインストールイメージへのパスです。

```
# luupgrade -n newbe -u -s /net/server/export/Solaris_10/combined.solaris_wos
```

6. (オプション) ブート環境がブート可能であることを確認します。

`lustatus` コマンドは、ブート環境の作成が完了したかどうかを報告します。

# 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. 新しいブート環境をアクティブにします。

```
# luactivate newbe
# init 6
```

ブート環境 newbe がアクティブになりました。

8. (オプション)別のブート環境にフォールバックします。新しいブート環境が実行可能でない場合、または別のブート環境に切り替える必要がある場合は、[第6章「障害回復:元のブート環境へのフォールバック\(タスク\)」](#)を参照してください。

非大域ゾーンが含まれているブート環境の管理

次のセクションでは、非大域ゾーンが含まれているブート環境の管理について説明します。

▼ ブート環境の非大域ゾーンのファイルシステムの構成を表示する

大域ゾーンと非大域ゾーンの両方のファイルシステムの一覧を表示するには、次の手順を実行します。

- ファイルシステムの一覧を表示します。

```
# lufslist -n BE_name
```

BE_name ファイルシステムの詳細を表示するブート環境の名前を指定します。

例 8-1 非大域ゾーンが含まれているファイルシステムの一覧を表示する

次の例では、非大域ゾーンが含まれているファイルシステムの一覧が表示されます。

```
# lufslist -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	-	

▼ 非大域ゾーンがインストールされているシステムのブート環境を比較する

lucompare コマンドは、非大域ゾーンの内容が含まれているブート環境の比較を行うようになりました。

- 現在のブート環境と新しいブート環境を比較します。

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

-i *infile* *infile* 中に指定されたファイルを比較します。比較するファイルは、絶対パスで指定する必要があります。ファイルのエントリがディレクトリである場合、比較はディレクトリに対して再帰的に行われます。このオプションまたは -t のいずれか一方を使用できます (両方は使用できません)。

-t バイナリ以外のファイルだけを比較します。この比較では、ファイルごとに file(1) コマンドを使用してそのファイルがテキストファイルであるかを確認します。ユーザーは、このオプションまたは -i のいずれか一方を使用できます (両方は使用できません)。

-o *outfile* 相違についての出力を *outfile* にリダイレクトします。

BE_name アクティブブート環境と比較するブート環境の名前を指定します。

例 8-2 ブート環境の比較

この例では、現在のブート環境 (ソース) と second_disk ブート環境が比較され、結果がファイルに出力されます。

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

非大域ゾーンが含まれるシステムで **lumount** コマンドを使用する

lumount コマンドは、非大域ゾーンが、非アクティブブート環境にある対応するファイルシステムにアクセスできるようにします。大域ゾーン管理者が **lumount** コマンドを使って非アクティブブート環境をマウントすると、同様にブート環境が非大域ゾーン用にマウントされます。

次の例では、ブート環境 **newbe** に対して、適切なファイルシステムが大域ゾーンの **/mnt** にマウントされます。実行中の非大域ゾーン、マウントされた非大域ゾーン、または準備完了の非大域ゾーンについては、**newbe** 内にある対応するファイルシステムが各ゾーン内の **/mnt** でも使用できるようになります。

```
# lumount -n newbe /mnt
```

マウントの詳細は、[lumount\(1M\)](#) のマニュアルページを参照してください。

Live Upgrade の例

この章では、新しいブート環境を作成、アップグレード、およびアクティブ化して、これを新たな稼動環境にする例を示します。

注 - この章では、UFS ファイルシステム用の Live Upgrade について説明します。UFS ファイルシステムを ZFS ルートプールに移行する、または ZFS ルートプールを作成およびインストールする手順については、[第 12 章「ZFS ルートプールのブート環境の作成」](#)を参照してください。

この章には次のセクションが含まれています。

- [159 ページの「Live Upgrade によるアップグレードの例」](#)
- [168 ページの「RAID-1 ボリューム \(ミラー\) の一方を切り離してアップグレードする例」](#)
- [172 ページの「既存のボリュームから Solaris ボリュームマネージャー RAID-1 ボリュームへ移行する例」](#)
- [172 ページの「空のブート環境を作成してフラッシュアーカイブをインストールする例」](#)

Live Upgrade によるアップグレードの例

この例では、Solaris 9 リリースを実行しているシステムで `lucreate` コマンドを使用して新しいブート環境を作成します。この新しいブート環境を、`luupgrade` コマンドを使用して Oracle Solaris 10 8/11 リリースにアップグレードします。アップグレードしたブート環境を、`luactivate` コマンドによってアクティブにします。この節では、以前のブート環境に戻す例も示します。

Live Upgrade を使用するための準備

Live Upgrade をはじめて実行する前に、インストールメディアから最新の Live Upgrade パッケージをインストールし、My Oracle Support ナレッジドキュメント 1004881.1 – Live Upgrade Software Patch Requirements (以前の 206844) に記載されているパッチをインストールする必要があります。My Oracle Support Web サイトでナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) を検索してください。

最新のパッケージとパッチにより、リリースに最新のバグ修正と新機能がすべて含まれるようになります。新しいブート環境の作成に進む前に、システムに関連するすべてのパッチを必ずインストールしてください。

次の手順は、My Oracle Support ナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) の手順を説明したものです。

注- この手順では、システムでボリュームマネージャーを実行していると仮定しています。Volume Manager を使用したリムーバブルメディアの管理の詳細は、『Oracle Solaris の管理: デバイスとファイルシステム』を参照してください。

1. スーパーユーザーになるか、同等の役割を引き受けます。

注- 役割には、認証と特権コマンドが含まれます。役割の詳細については、『Solaris のシステム管理: セキュリティーサービス』の「RBAC の構成 (タスクマップ)」を参照してください。

2. My Oracle Support Web サイトで、ナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) の手順に従って、Live Upgrade パッケージを削除して追加します。

- a. 既存の Live Upgrade パッケージを削除します。

3 つの Live Upgrade パッケージ SUNWluu、SUNWlur、および SUNWlucfg は、Live Upgrade を使ってアップグレードまたはパッチを適用するのに必要なソフトウェアを構成します。これらのパッケージには、既存のソフトウェア、新しい機能、およびバグ修正が含まれています。Live Upgrade を使用する前に、既存パッケージを削除しないで新しいパッケージをシステムにインストールすると、ターゲットリリースへのアップグレードまたはパッチ適用は失敗します。SUNWlucfg パッケージは、Solaris 10 8/07 以降のリリースの新機能です。Oracle Solaris 10 8/07 より前のリリースの Live Upgrade パッケージを使用する場合は、このパッケージを削除する必要はありません。

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- b. 新しい Live Upgrade パッケージをインストールします。

パッケージをインストールするには、インストール DVD または CD にある `liveupgrade20` コマンドか、`pkgadd` コマンドを使用します。`liveupgrade20` コマンドを使用する場合は、Java ソフトウェアが必要です。システムに Java ソフトウェアがインストールされていない場合は、`pkgadd` コマンドを使用してパッケージをインストールする必要があります。詳細は、My Oracle Support ナレッジドキュメントを参照してください。

- Oracle Solaris Operating System DVD を使用している場合は、ディレクトリを変更してインストーラを実行します。
 - ディレクトリを変更します。

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

注 - SPARC ベースのシステムの場合、Solaris 10 10/08 リリースより前のリリースではインストーラへのパスが異なります。

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

- インストーラを実行します。

```
# ./liveupgrade20 -noconsole - nodisplay
```

`-noconsole` オプションと `-nodisplay` オプションを使用すると、キャラクターユーザーインタフェース (character user interface、CUI) が表示されるのを防ぐことができます。

注 - Live Upgrade CUI はサポートされなくなりました。

- Oracle Solaris Software - 2 CD を使用する場合は、パスを変更せずにインストーラを実行できます。

```
% ./installer
```

- パッケージが正常にインストールされていることを確認します。

```
# pkgchk -v SUNWlucfg SUNWlur SUNWluu
```

3. ナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) に記載されているパッチをインストールします。

- a. ローカルディスクにパッチを保存する場合は、`/var/tmp/lupatches` などのディレクトリを作成します。
- b. My Oracle Support の Web サイトから、パッチの一覧を取得します。
- c. この例のようにパッチディレクトリに移動します。

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

- d. パッチをインストールします。

```
# patchadd -M path-to-patchespatch-id patch-id
```

path-to-patches は、*/var/tmp/lupatches* などのパッチディレクトリへのパスです。*patch_id* はパッチの番号です。複数のパッチ名を指定する場合は、スペースで区切ります。

注 - パッチはナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) に指定されている順序で適用する必要があります。

- e. 必要に応じてシステムをリブートします。いくつかのパッチは、有効にするためにリブートする必要があります。

x86 のみ: システムをリブートする必要があります。それ以外の場合、Live Upgrade が失敗します。

```
# init 6
```

以上で新しいブート環境を正しく作成するために必要なパッケージとパッチがインストールされました。

PreFlightChecker ツールの使用

Live Upgrade の PreFlightChecker ツールを使用すると、Live Upgrade の動作の失敗を招くことがあるシステム構成の問題を検出できます。このツールは、**lucreate** コマンドを起動してブート環境を作成する前に実行しておく必要があります。

次の表に、PreFlightChecker ツールを使用してシステムチェックを実行し、あるファイルシステムから別のファイルシステムに移行する方法を示します。

タスク	使用例	出力例
システムチェックを実行する	# lupc -s	This system has Patch level/IDR of IDR147654-04. It is advisable to apply the latest Live Upgrade patch from MOS.
UFS ファイルシステムから別のファイルシステムに移行する	# lupc --ufs2ufs -m /:/dev/dsk/c1t5d0s0:ufs	This system has Patch level/IDR of 121430-75 It is advisable to apply the latest Live Upgrad patch from MOS. File system has sufficient space to accommodate ABE Required space is 4995481 KB and fs has 4131006 KB

タスク	使用例	出力例
ZFS ファイルシステムから別のファイルシステムに移行する	<code># lupc -zfs2zfs -p newpool</code>	<p>This system has Patch level/IDR of IDR147654-04 It is advisable to apply the latest Live Upgradepatch from MOS.</p> <p>File system has sufficient space to accommodate ABE Required space is 20787.2 KB and fs has 10422845.44 KB</p>
UFS ファイルシステムから ZFS ファイルシステムに移行する	<code># lupc --ufs2zfs -p bepool -D /var</code>	<p>This system has Patch level/IDR of 121430-75 It is advisable to apply the latest Live Upgrade patch from MOS.</p> <p>File system has sufficient space to accommodate ABE Required space is 4995481 KB and fs has 20761804.8 KB</p> <p>File system has sufficient space to accommodate ABE Required space is 5610220 KB and fs has 20761804.8 KB</p>

ブート環境を作成する方法

-c オプションを使用して、ソースブート環境に `c0t4d0s0` という名前を付けます。ソースブート環境の名前設定は最初のブート環境を作成するときだけ必要です。-c オプションによる命名の詳細は、「ブート環境をはじめて作成する方法」の[手順 1](#)を参照してください。

新しいブート環境の名前は `c0t15d0s0` です。-A オプションを使用して、このブート環境名に関連付ける説明を作成します。

ルート (/) ファイルシステムを新しいブート環境にコピーします。また、ソースブート環境のスワップスライスは共有せずに、新しいスワップスライスを作成します。

```
# lucreate -A 'BE_description' -c /dev/dsk/c0t4d0s0 -m /dev/dsk/c0t15d0s0:ufs\
-m -:/dev/dsk/c0t15d0s1:swap -n /dev/dsk/c0t15d0s0
```

非アクティブブート環境をアップグレードする方法

非アクティブブート環境の名前は `c0t15d0s0` です。アップグレードに使用するオペレーティングシステムイメージはネットワークから取得します。

```
# luupgrade -n c0t15d0s0 -u -s /net/ins-svr/export/Solaris_10 \
combined.solaris_wos
```

ブート環境がブート可能か確認する方法

lustatus コマンドは、ブート環境の作成が完了したかどうかを報告します。lustatus コマンドは、ブート環境がブート可能であるかどうかも報告します。

```
# lustatus
boot environment  Is      Active  Active  Can      Copy
Name             Complete Now    OnReboot Delete   Status
-----
c0t4d0s0         yes     yes     yes     no      -
c0t15d0s0         yes     no      no      yes     -
```

非アクティブブート環境をアクティブにする方法

luactivate コマンドを使用して c0t15d0s0 ブート環境をブート可能にします。続いてシステムをリブートします。これで c0t15d0s0 がアクティブブート環境になります。c0t4d0s0 ブート環境は非アクティブになります。

```
# luactivate c0t15d0s0
# init 6
```

(オプション)ソースブート環境へ戻す方法

新しいブート環境のアクティブ化の状況に応じて、フォールバックのための手順を次の中から選択します。

- SPARC ベースシステムの場合:
 - アクティブ化が正常に行われたが元のブート環境に戻すという場合は、[例 9-1](#)を参照してください。
 - アクティブ化に失敗したが、元のブート環境からブート可能な場合は、[例 9-2](#)を参照してください。
 - アクティブ化に失敗し、メディアまたはネットインストールイメージを使用して元のブート環境に戻る必要がある場合は、[例 9-3](#)を参照してください。
- x86 ベースシステムで、**Solaris 10 1/06** 以降のリリースの場合および Oracle Solaris の機能である GRUB メニューを使用する場合:
 - アクティブ化に失敗し、GRUB メニューは適切に表示されるが新しいブート環境のブートに失敗する場合は、[例 9-4](#)を参照してください。
 - アクティブ化に失敗し、GRUB メニューが表示されない場合は、[例 9-5](#)を参照してください。

例 9-1 SPARC: ブート環境作成は正常に完了したが元のブート環境に戻す場合

この例では、新しいブート環境のアクティブ化が正常に完了したにもかかわらず、元の `c0t4d0s0` ブート環境をアクティブブート環境として復元しています。デバイス名は `first_disk` です。

```
# /sbin/luactivate first_disk
# init 6
```

例 9-2 SPARC: ブート環境のアクティブ化に失敗した場合のフォールバック

この例では、新しいブート環境のブートに失敗しています。シングルユーザーモードで元のブート環境 `c0t4d0s0` からブートさせるために、OK プロンプトを表示させる必要があります。

```
OK boot net -s
# /sbin/luactivate first_disk
Do you want to fallback to activate boot environment c0t4d0s0
(yes or no)? yes
# init 6
```

元のブート環境 `c0t4d0s0` がアクティブブート環境になります。

例 9-3 SPARC: DVD、CD、またはネットワークインストールイメージを使って元のブート環境に戻す

この例では、新しいブート環境のブートに失敗しています。元のブート環境からはブートできないためメディアまたはネットインストールイメージを使用する必要があります。デバイスは `/dev/dsk/c0t4d0s0` です。元のブート環境 `c0t4d0s0` がアクティブブート環境になります。

```
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
```

例 9-4 x86: GRUB メニューを使用して元のブート環境に戻す

次の例では、**Solaris 10 1/06** 以降のリリースで、GRUB メニューを使用して戻す手順を示します。

この例では、GRUB メニューは適切に表示されますが、新しいブート環境のブートに失敗します。フォールバックを可能にするため、元のブート環境はシングルユーザーモードでブートされます。

1. スーパーユーザーになるか、同等の役割を引き受けます。

例 9-4 x86: GRUB メニューを使用して元のブート環境に戻す (続き)

注-役割には、認証と特権コマンドが含まれます。役割の詳細については、『[Solaris のシステム管理: セキュリティーサービス](#)』の「RBAC の構成 (タスクマップ)」を参照してください。

- GRUB メニューを表示するにはシステムをリブートします。

```
# init 6
```

GRUB メニューが表示されます。

```
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.
```

- GRUB メニューから、元のブート環境を選択します。ブート環境は、GRUB ソフトウェアで作成されている必要があります。**Solaris 10 1/06** リリースより前に作成されたブート環境は、GRUB ブート環境ではありません。ブート可能な GRUB ブート環境がない場合は、[例 9-5](#)に進んでください。
- 次のように入力して、GRUB メニューを編集します。e。
- 矢印キーを使用して kernel /boot/multiboot を選択し、e と入力します。GRUB 編集メニューが表示されます。

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

- s と入力して、シングルユーザーモードでブートします。

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

- ブートして、ブート環境をマウントします。次に、それをアクティブにします。

```
# 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
```

例 9-5 x86: DVD または CD を使用して GRUB メニューで元のブート環境に戻す

次の例では、**Solaris 10 1/06** 以降のリリースにおいて DVD または CD を使用して戻す手順を示します。

例 9-5 x86: DVD または CD を使用して GRUB メニューで元のブート環境に戻す (続き)

この例では、新しいブート環境のブートに失敗しています。また、GRUB メニューは表示されません。フォールバックを可能にするため、元のブート環境はシングルユーザーモードでブートされます。

1. Oracle Solaris Operating System DVD (x86 版) または Oracle Solaris Software for x86 Platforms - 1 CD を挿入します。
2. スーパーユーザーになるか、同等の役割を引き受けます。

注-役割には、認証と特権コマンドが含まれます。役割の詳細については、『Solaris のシステム管理: セキュリティーサービス』の「RBAC の構成(タスクマップ)」を参照してください。

3. DVD または CD からブートします。

```
# init 6
```

GRUB メニューが表示されます。

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

```
+-----+
|Oracle Solaris 10 8/11                                     |
|Oracle Solaris 10 8/11 Serial Console ttya                |
|Oracle Solaris 10 8/11 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. デフォルトのオプションがブートされるまで待機するか、表示されているオプションのどれかを選択します。

インストール画面が表示されます。

```
+-----+
|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」オプションを選択します。

例 9-5 x86: DVD または CD を使用して GRUB メニューで元のブート環境に戻す (続き)

次のメッセージが表示されます。

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

6. 次のコマンドを入力します。

```
Starting shell...
```

```
#
```

これでシングルユーザーモードになりました。

7. ブート環境をマウントします。次に、アクティブにしてリブートします。

```
# 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 ボリューム(ミラー)の一方を切り離してアップグレードする例

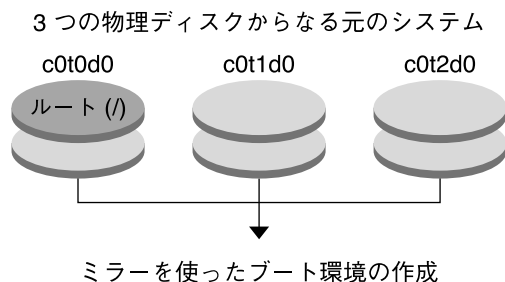
この例では、次のタスクの手順を示します。

- 新しいブート環境に RAID-1 ボリューム(ミラー)を作成します
- ミラーの一方を切り離し、アップグレードします
- ミラー(連結)の他方を新しいミラーに接続します

図 9-1 は、3 つの物理ディスクから成る現在のブート環境を示します。

図 9-1 RAID-1 ボリューム(ミラー)の一方を切り離し、アップグレードする

RAID-1 ボリューム(ミラー)の一方を切り離し、アップグレードする



1. ミラーを持つ新しいブート環境 `second_disk` を作成します。

次のコマンドは、次のようなタスクを実行します。

- `lucreate` コマンドにより、ルート (/) マウントポイントの UFS ファイルシステムが構成されます。d10 というミラーが作成されます。このミラー d10 に、現在のブート環境のルート (/) ファイルシステムがコピーされます。ミラー d10 にあるデータはすべて上書きされます。
- 2つのスライス `c0t1d0s0` および `c0t2d0s0` は、サブミラーとして指定されています。これら2つのサブミラーは、ミラー d10 に接続されます。

```
# 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. ブート環境 `second_disk` をアクティブにします。

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

3. 別のブート環境 `third_disk` を作成します。

次のコマンドは、次のようなタスクを実行します。

- `lucreate` コマンドにより、ルート (/) マウントポイントの UFS ファイルシステムが構成されます。d20 というミラーが作成されます。
- スライス `c0t1d0s0` が現在のミラーから切り離され、ミラー d20 に追加されます。このサブミラーの内容であるルート (/) ファイルシステムは保持され、コピー処理は発生しません。

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

4. 新しいブート環境 `third_disk` をアップグレードします。

```
# luupgrade -u -n third_disk \
-s /net/installmachine/export/Solaris_10/OS_image
```

5. アップグレードされたブート環境にパッチを追加します。

```
# luupgrade -t n third_disk -s /net/patches 222222-01
```

6. ブート環境 `third_disk` をアクティブにして、このブート環境からシステムを実行します。

```
# /sbin/luactivate third_disk
# init 6
```

7. ブート環境 `second_disk` を削除します。

```
# ludelete second_disk
```

8. 次のコマンドは、次のようなタスクを実行します。

- ミラー d10 を消去します。
- `c0t2d0s0` の連結の数を調べます。

- `metastat` コマンドで見つけた連結を、ミラー `d20` に接続します。`metattach` コマンドは、新しく接続した連結と、ミラー `d20` の連結とを同期します。連結にあるデータはすべて上書きされます。

```
# metaclear d10
# metastat -p | grep c0t2d0s0
dnum 1 1 c0t2d0s0
# metattach d20 dnum
```

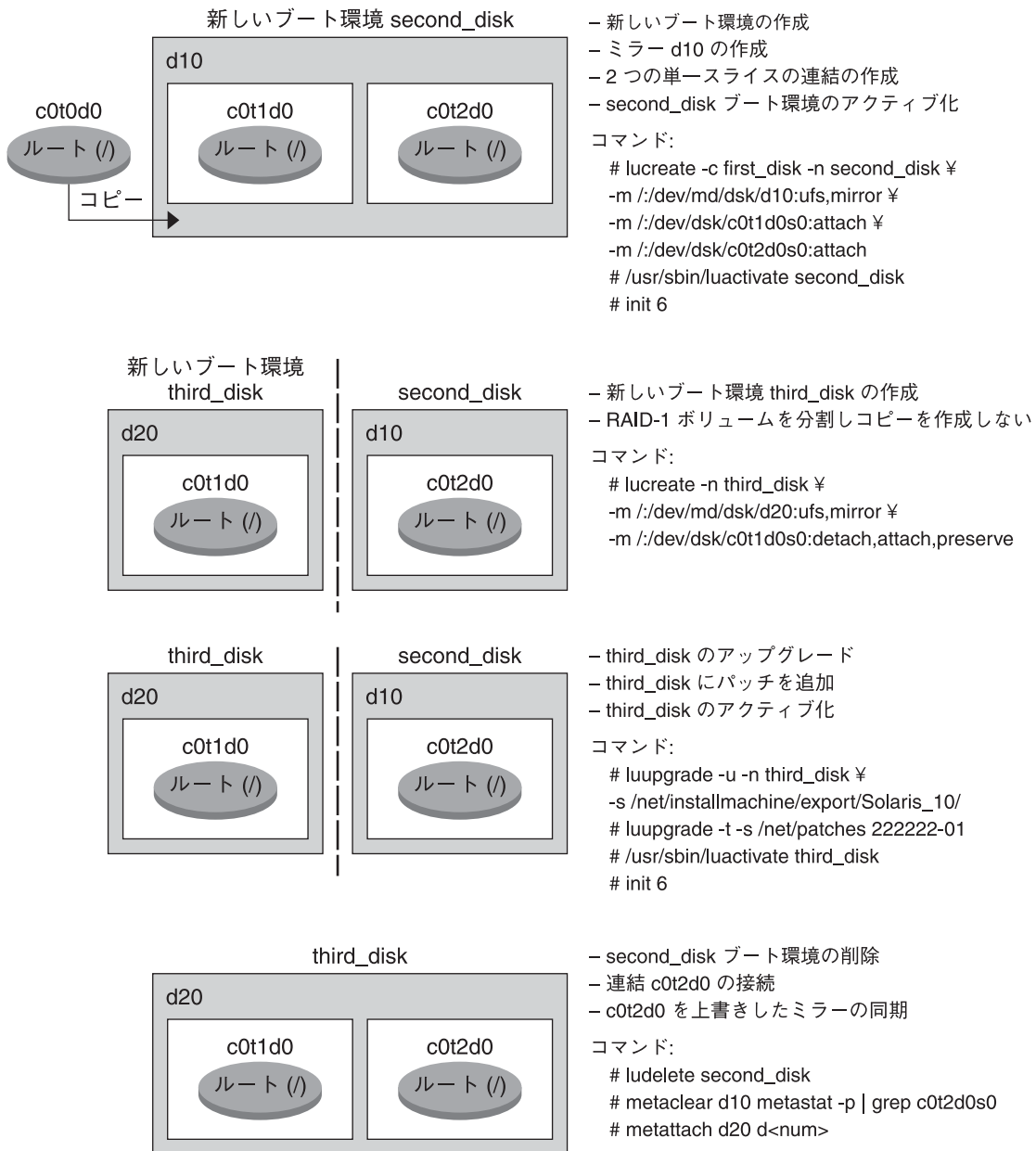
`num` `metastat` コマンドで見つかった連結の数。

これで、新しいブート環境 `third_disk` がアップグレードされ、この環境からシステムが実行されます。`third_disk` には、ミラー化されたルート (`/`) ファイルシステムが含まれています。

図 9-2 は、前の例のコマンドでミラーを切り離してアップグレードする手順の全体を示しています。

図 9-2 RAID-1 ボリューム(ミラー)の一方を切り離し、アップグレードする(続き)

RAID-1 ボリューム(ミラー)の一方を切り離し、アップグレードする(続き)



既存のボリュームから **Solaris** ボリュームマネージャー RAID-1 ボリュームへ移行する例

Live Upgrade では、RAID-1 ボリューム (ミラー) 上に新しいブート環境を作成できます。現在のブート環境のファイルシステムは、次のいずれかです。

- 物理ストレージデバイス
- Solaris ボリュームマネージャーの制御下の RAID-1 ボリューム
- Veritas VXFES 制御下のボリューム

ただし、新しいブート環境のターゲットは、Solaris ボリュームマネージャー RAID-1 ボリュームでなければなりません。たとえば、ルート(/) ファイルシステムのコピー用に指定するスライスは、/dev/vx/dsk/rootvol となります。rootvol はルート(/) ファイルシステムを含むボリュームです。

この例では、現在のブート環境のルート(/) ファイルシステムは Solaris ボリュームマネージャーボリューム以外のボリューム上にあります。新しいブート環境では、Solaris ボリュームマネージャー RAID-1 ボリュームである c0t2d0s0 上にルート(/) ファイルシステムが作成されます。lucreate コマンドは、現在のボリュームを Solaris ボリュームマネージャーボリュームに移行させるコマンドです。新しいブート環境の名前は svm_be です。lustatus コマンドを使用すると、新しいブート環境のアクティブ化とリブートの準備ができていいるかどうかわかります。ブート環境がアクティブにされ、現在のブート環境になります。

```
# lucreate -n svm_be -m /:/dev/md/dsk/d1:mirror,ufs \
-m /:/dev/dsk/c0t2d0s0:attach
# lustatus
# luactivate svm_be
# lustatus
# init 6
```

空のブート環境を作成してフラッシュアーカイブをインストールする例

次の手順を3段階に分けて説明します。

- 空のブート環境を作成する
- アーカイブをインストールする
- ブート環境をアクティブにし、現在実行中のブート環境にする

lucreate コマンドは、アクティブブート環境内のファイルシステムに基づいてブート環境を作成します。lucreate コマンドに -s- オプションを指定して実行すると、空のブート環境を短時間で作成できます。スライスは、指定のファイルシステム用に予約されていますが、ファイルシステムはコピーされません。このブート環

境は、名前が付けられますが、フラッシュアーカイブがインストールされるまでは実際には作成されません。空のブート環境にアーカイブがインストールされると、ファイルシステムは予約されたスライスにインストールされます。その後、ブート環境をアクティブにします。

空のブート環境を作成する方法

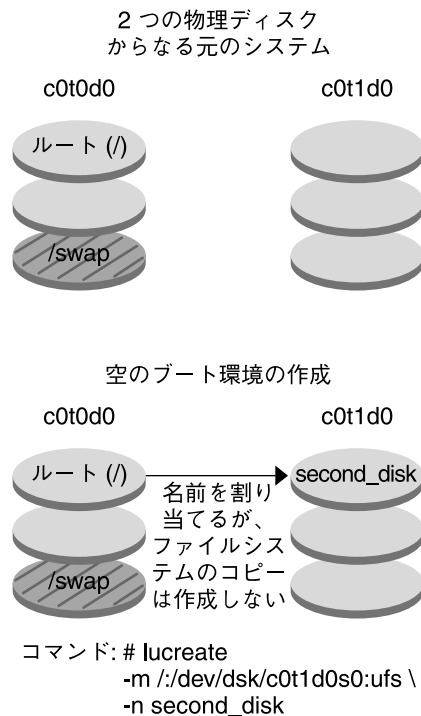
最初の手順で、空のブート環境を作成します。指定されたファイルシステム用にスライスが予約されますが、現在のブート環境からファイルシステムがコピーされることはありません。新しいブート環境の名前は `second_disk` です。

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

これで、ブート環境にフラッシュアーカイブを格納する準備ができました。

図 9-3 は、空のブート環境の作成の様子を示しています。

図 9-3 空のブート環境の作成



新しいブート環境へフラッシュアーカイブをインストールする方法

2 番目の手順では、前の例で作成した `second_disk` ブート環境に、アーカイブをインストールします。アーカイブはローカルシステムに存在します。`-s` および `-a` オプションで指定するオペレーティングシステムのバージョンは、どちらも Oracle Solaris 10 8/11 リリースです。アーカイブの名前は `Solaris_10.flar` です。

```
# luupgrade -f -n second_disk \  
-s /net/installmachine/export/Solaris_10/OS_image \  
-a /net/server/archive/10.flar
```

このようにして、ブート環境をアクティブにする準備が整います。

新しいブート環境をアクティブにする方法

最後の手順では、`luactivate` コマンドを使用して、`second_disk` ブート環境をブート可能な状態にします。続いてシステムをリブートします。これで `second_disk` がアクティブブート環境になります。

```
# luactivate second_disk  
# init 6
```

- 空のブート環境の作成手順については、[81 ページの「フラッシュアーカイブ用の空のブート環境を作成する方法」](#)を参照してください。
- フラッシュアーカイブの作成手順については、『[Oracle Solaris 10 1/13 インストールガイド: フラッシュアーカイブ \(作成とインストール\)](#)』の第 3 章「[フラッシュアーカイブの作成 \(タスク\)](#)」を参照してください。
- ブート環境をアクティブ化したり、元のブート環境に戻したりする詳細な手順については、[第 6 章「障害回復: 元のブート環境へのフォールバック \(タスク\)」](#)を参照してください。

パート II

Live Upgrade を使った ZFS ルートプールの のアップグレードと ZFS ルートプール への移行

このパートでは、Live Upgrade を使って ZFS ストレージプールに非アクティブ
ブート環境を作成し、アップグレードするための概要と手順を説明します。UFS
ルート (/) ファイルシステムを ZFS ルートプールに移行することもできます。

Live Upgrade と ZFS (概要)

Live Upgrade を使用して、UFS ファイルシステムを ZFS ルートプールに移行し、既存の ZFS ルートプールから ZFS ルートファイルシステムを作成できます。

注 - Live Upgrade を使用したブート環境の作成は、Solaris 10 10/08 リリースの新機能です。UFS ファイルシステムに対する Live Upgrade の実行時には、コマンド行パラメータにも Live Upgrade の操作にも変更はありません。UFS ファイルシステムがインストールされているシステムで Live Upgrade を実行する方法については、このドキュメントの [パート I 「Live Upgrade を使ったアップグレード」](#) を参照してください。

次のセクションで、これらのタスクの概要を説明しています。

- [179 ページの「ZFS での Live Upgrade の使用の概要」](#) .
- SVM ボリュームを使用した (または使用しない) UFS ファイルシステムの移行。
 - [180 ページの「UFS ファイルシステムから ZFS ルートプールへの移行」](#)
 - [183 ページの「Solaris ボリュームマネージャーボリュームで構成された UFS ファイルシステムの ZFS ルートファイルシステムへの移行」](#)
- 既存のルートプール内または別のルートプールでの新しい ZFS ブート環境の作成。
 - [184 ページの「同一のルートプール内での新規ブート環境の作成」](#)
 - [186 ページの「別のルートプール上への新規ブート環境の作成」](#)
- [188 ページの「現在稼働中のシステム以外のソースからの新規ブート環境の作成」](#)
- [188 ページの「非大域ゾーンがインストールされているシステム上への ZFS ブート環境の作成」](#)

Oracle Solaris 10 8/11 リリースの新機能

Oracle Solaris 10 8/11 以降のリリースでは、ZFS ファイルシステムに次のインストール拡張機能があります。

- Live Upgrade の `luupgrade` コマンドを使用して、フラッシュアーカイブを ZFS ルート代替ブート環境にインストールできます。

以前のリリースでの ZFS フラッシュアーカイブとは異なり、ZFS ルートマスターシステム上に作成されたフラッシュアーカイブには、既存のブート環境がすべて含まれるわけではありません。代わりに、そのアーカイブにはアクティブな ZFS ブート環境のみが含まれます。そのアーカイブには、`lucreate` コマンドの `-D` オプションを使って明示的に除外されるデータセットと、最上位のプールのデータセット内にあるユーザーデータは含まれません。スワップボリュームおよびダンプボリュームはそのアーカイブに含められませんが、フラッシュアーカイブのインストール時に作成されます。

ZFS フラッシュアーカイブの作成およびインストールの詳細については、『[Oracle Solaris ZFS 管理ガイド](#)』の「[ZFS ルートファイルシステムのインストール \(Oracle Solaris フラッシュアーカイブインストール\)](#)」を参照してください。

- Live Upgrade の `lucreate` コマンドの `-D` オプションを使用して、UFS ルートファイルシステムを ZFS ルートファイルシステムに移行するときに `/var` 用の別個のデータセットを作成できます。

ただし、別の OS クリティカルファイルシステムについては、`-D` オプションを使用して別個のデータセットを指定することはできません。`/var` ファイルシステムは、ZFS ルートブート環境内で単独のデータセットとすることが許可される、唯一の OS クリティカルファイルシステムです。`-D` オプションは、ソースブート環境が UFS ルートで、作成されるターゲットブート環境が ZFS ルートの場合にのみ許可されます。

詳細は、[180 ページ](#)の「[UFS ルート \(/\) ファイルシステムから ZFS ルートプールへの移行](#)」を参照してください。

手順と制限事項の詳細は、『[Oracle Solaris ZFS 管理ガイド](#)』の第 4 章「[Oracle Solaris ZFS ルートファイルシステムのインストールとブート](#)」を参照してください。

Solaris 10 10/09 リリースの新機能

Solaris 10 10/09 リリース以降では、JumpStart プロファイルを設定して、ZFS ルートプールのフラッシュアーカイブを特定できます。

フラッシュアーカイブは、UFS ルートファイルシステムまたは ZFS ルートファイルシステムを実行しているシステムで作成できます。ZFS ルートプールのフラッシュアーカイブには、スワップボリュームとダンプボリュームおよび任意の除

外されたデータセットを除く、プール階層全体が含まれます。スワップボリュームとダンプボリュームは、フラッシュアーカイブのインストール時に作成されます。

フラッシュアーカイブによるインストール方法は次のとおりです。

- ZFS ルートファイルシステムによるシステムのインストールとブートに使用できるフラッシュアーカイブを生成します。
- ZFS フラッシュアーカイブを使用して、システムの JumpStart インストールを実行します。

注-ZFS フラッシュアーカイブを作成すると、個別のブート環境ではなく、ルートプール全体がバックアップされます。flarcreate コマンドと flar コマンドの -D オプションを使用すると、プール内の個別のデータセットを除外できます。

詳細と制限事項については、『Oracle Solaris ZFS 管理ガイド』の「ZFS ルートファイルシステムのインストール (Oracle Solaris フラッシュアーカイブインストール)」を参照してください。

ZFS での Live Upgrade の使用の概要

UFS ファイルシステムがある場合は、Live Upgrade は以前のリリースと同様に動作します。今回のリリースでは、UFS ファイルシステムから ZFS ルートプールへの移行、および ZFS ルートプール内での新規ブート環境の作成が可能になりました。これらのタスク向けに、lucreate コマンドが -p オプションで拡張されました。lucreate コマンドには次の構文があります。

```
# lucreate [-c active-BE-name] -n BE-name [-p zfs-root-pool [-D /var]]
```

-p オプションでは、新しいブート環境を配置する ZFS プールを指定します。このオプションは、ソースブート環境とターゲットブート環境が同一のプール内にある場合は省略できます。

lucreate コマンドの -m オプションは ZFS ではサポートされません。その他の lucreate コマンドのオプションは、一部の例外を除いてこれまでと変わりありません。制限事項については、189 ページの「Live Upgrade 使用時のシステム要件と制限事項」を参照してください。

ZFS についての概要、計画、手順などの情報は、『Oracle Solaris ZFS 管理ガイド』を参照してください。

UFS ファイルシステムから ZFS ルートプールへの移行

現在稼働中のシステムからブート環境を作成する場合は、`lucreate` コマンドで UFS ルート (/) ファイルシステムを ZFS ルートプールにコピーします。コピー処理には、システムによって時間がかかる場合があります。

UFS ファイルシステムから移行する場合は、ディスクスライス上の UFS ルート (/) ファイルシステムをソースブート環境にすることができます。ZFS ルートプール上のソースブート環境から UFS ファイルシステム上のブート環境を作成することはできません。

UFS ルート (/) ファイルシステムから ZFS ルートプールへの移行

このセクションで説明するコマンドは、ZFS ルートプールを作成し、UFS ルート (/) ファイルシステムから ZFS ルートプール内に新しいブート環境を作成します。ZFS ルートプールは `lucreate` を使用する前に作成します。また、アップグレード可能かつブート可能にするために、ディスク全体ではなくスライス上に作成します。ディスクには、EFI ラベルでなく SMI ラベルが必要です。その他の制限事項については、[189 ページの「Live Upgrade 使用時のシステム要件と制限事項」](#)を参照してください。

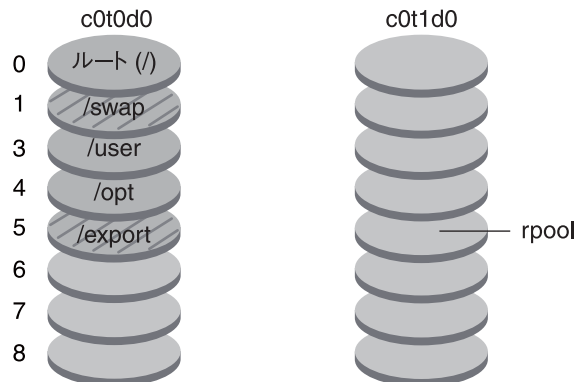
次の図に、ルートプール `rpool` を別のスライス `c0t1d0s5` 上に作成する `zpool` コマンドを示します。ディスクスライス `c0t0d0s0` には、UFS ルート (/) ファイルシステムが含まれています。`lucreate` コマンドの `-c` オプションで指定する現在稼働中のシステム `c0t0d0` が、UFS ルート (/) ファイルシステムです。`-n` オプションで、作成するブート環境に `new-zfsBE` という名前を割り当てます。`-p` オプションでは、新しいブート環境を `rpool` に配置することを指定します。UFS の `/export` ファイルシステムと `/swap` ボリュームは、新しいブート環境にコピーされません。

図 10-1 UFS ファイルシステムから ZFS ルートプールへの移行

UFS ファイルシステムから **ZFS** ルートプールへの移行

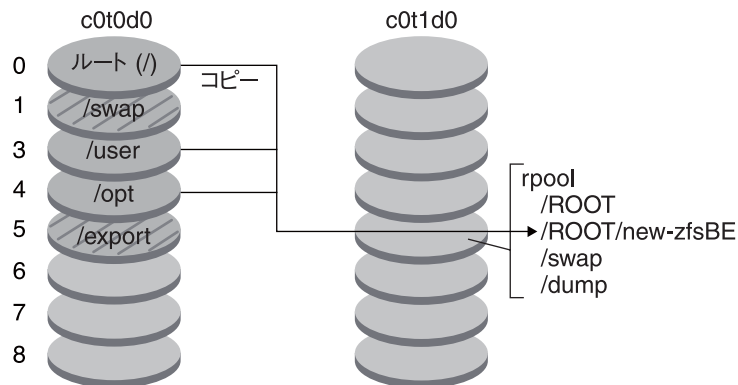
- ① スライス上に
- ZFS**
- ルートプールを作成する。

```
# zpool create rpool c0t1d0s5
```



- ②
- UFS**
- ブート環境を
- rpool**
- にコピーする。

```
# lucreate -c c0t0d0 -n new-zfsBE -p rpool
```



代替ブート環境内で `/var` についての単独のデータセットの作成を指定するには、`lucreate` コマンドの `-D` オプションを使用します。

```
lucreate -c c0t0d0 -n new-zfsBE -p rpool -D /var
```

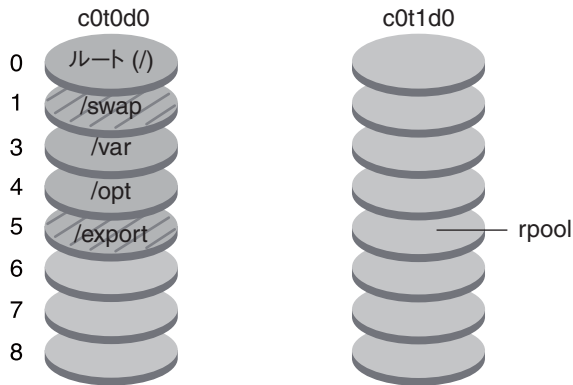
次の図に、このサンプルの `lucreate` コマンドの一部として `rpool` 内に作成されるデータセットを示します。

図 10-2 UFS ファイルシステムから ZFS ルートプールへの移行

UFS ファイルシステムから ZFS ルートプールへの移行

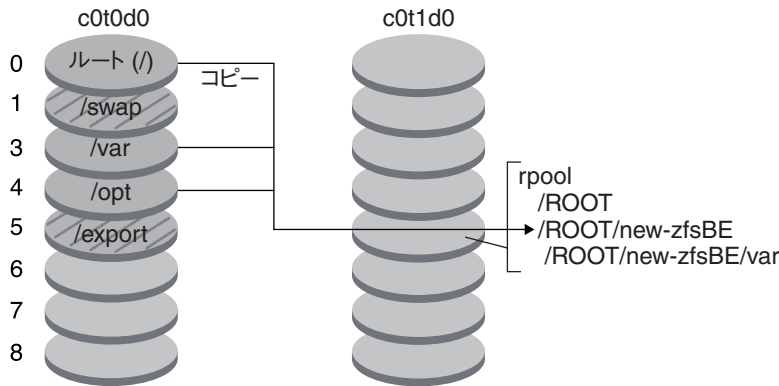
① UFS ファイルシステムから ZFS ルートプールへの移行スライス上に ZFS ルートプールを作成する。

```
# zpool create rpool c0t1d0s5
```



② UFS ブート環境を `rpool` にコピーする。

```
# lucreate -c c0t0d0 -n new-zfsBE -p rpool -D /var
```



`lucreate` コマンドで `-D /var` を指定しない場合、`/var` がソースブート環境内で単独のファイルシステムであっても、代替ブート環境内では `/var` についての単独のデータセットは作成されません。

例 10-1 UFS ルート (`/`) ファイルシステムから ZFS ルートプールへの移行

この例は、[図 10-1](#)と同じコマンドを示しています。これらのコマンドで、新規のルートプール `rpool` を作成し、UFS ルート (`/`) ファイルシステムからプール内に新規ブート環境を作成します。この例では、`zfs list` コマンドで、`zpool` コマンドによって作成した ZFS ルートプールを表示しています。次の `zfs list` コマンドでは、`lucreate` コマンドで作成されたデータセットを表示しています。

例 10-1 UFS ルート (/) ファイルシステムから ZFS ルートプールへの移行 (続き)

```
# zpool create rpool c0t1d0s5
# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool	5.97G	23.3G	31K	/rpool

```
# lucreate -c c0t0d0 -n new-zfsBE -p rpool
# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool	5.97G	23.3G	31K	/rpool
rpool/ROOT	4.42G	23.3G	31K	legacy
rpool/ROOT/new-zfsBE	4.42G	23.3G	4.42G	/
rpool/dump	1.03G	24.3G	16K	-
rpool/swap	530M	23.8G	16K	-

次の `zfs list` コマンドでは、`lucreate` コマンドの `-D /var` オプションを使用することによって、`/var` について作成された単独のデータセットを示しています。

```
# lucreate -c c0t0d0 -n new-zfsBE -p rpool -D /var
# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool	5.97G	23.3G	31K	/rpool
rpool/ROOT	4.42G	23.3G	31K	legacy
rpool/ROOT/new-zfsBE	4.42G	23.3G	4.42G	/
rpool/ROOT/new-zfsBE/var	248MG	23.3G	248M	/var
rpool/dump	1.03G	24.3G	16K	-
rpool/swap	530M	23.8G	16K	-

新しいブート環境は `/new-zfsBE` です。ブート環境 `new-zfsBE` は、すぐにアップグレードおよびアクティブにできます。

Solaris ボリュームマネージャーボリュームで構成された UFS ファイルシステムの ZFS ルートファイルシステムへの移行

Solaris ボリュームマネージャー (SVM) ボリュームがあるシステムでも、UFS ファイルシステムの移行が可能です。既存の SVM 構成から UFS ブート環境を作成するため、現在稼動中のシステムから新規ブート環境を作成します。その後、その新しい UFS ブート環境から ZFS ブート環境を作成します。

例 10-2 SVM ボリュームのある UFS ルート (/) ファイルシステムから ZFS ルートプールへの移行

SVM ボリュームのあるシステムを移行する場合、SVM ボリュームは無視されます。次の例に示すように、ルートプール内にミラーを設定できます。

この例では、`-m` オプションを指定した `lucreate` コマンドで、現在稼動中のシステムから新規ブート環境を作成します。ディスクスライス `c1t0d0s0` には、SVM ボリュームで構成された UFS ルート (/) ファイルシステムが含まれています。`zpool` コ

例 10-2 SVM ボリュームのある UFS ルート (/) ファイルシステムから ZFS ルートプールへの移行 (続き)

マンドで、ルートプール `c1t0d0s0`、および RAID-1 ボリューム(ミラー) `c2t0d0s0` を作成します。2 番目の `lucreate` コマンドの `-n` オプションで、作成するブート環境に `c0t0d0s0` という名前を割り当てます。`-s` オプションでは、UFS ルート (/) ファイルシステムを指定します。`-p` オプションでは、新しいブート環境を `rpool` に配置することを指定します。

```
# lucreate -n ufsBE -m /:/dev/md/dsk/d104:ufs
# zpool create rpool mirror c1t0d0s0 c2t1d0s0
# lucreate -n c0t0d0s0 -s ufsBE -p zpool
```

ブート環境 `c0t0d0s0` は、すぐにアップグレードおよびアクティブにできます。

ZFS ルートプールからの新規ブート環境の作成

新規 ZFS ブート環境は、同一ルートプール内、新規ルートプール上のどちらにも作成できます。このセクションには次の概要が含まれます。

- [184 ページの「同一のルートプール内での新規ブート環境の作成」](#)
- [186 ページの「別のルートプール上への新規ブート環境の作成」](#)

同一のルートプール内での新規ブート環境の作成

同一の ZFS ルートプール内で新規ブート環境を作成する場合は、`lucreate` コマンドでソースブート環境からスナップショットを作成し、そのスナップショットからクローンを作成します。スナップショットとクローンの作成はきわめて短時間で完了し、ディスク容量の消費も最小限ですみます。最終的にどれくらいの容量が必要かは、アップグレード処理の一環として置き換えられるファイルの数で決まります。スナップショットは読み取り専用ですが、クローンはスナップショットの読み書き可能なコピーです。クローンブート環境に加えられた変更は、スナップショットにも、スナップショットの作成元のソースブート環境にも反映されません。

注-有効なデータセット内のデータが変更されると、スナップショットは古いデータを参照し続けるための領域を消費します。その場合、スナップショットのため、古いデータの領域は解放されずプールに戻されません。スナップショットの詳細については、『[Oracle Solaris ZFS 管理ガイド](#)』の第 6 章「[Oracle Solaris ZFS のスナップショットとクローンの操作](#)」を参照してください。

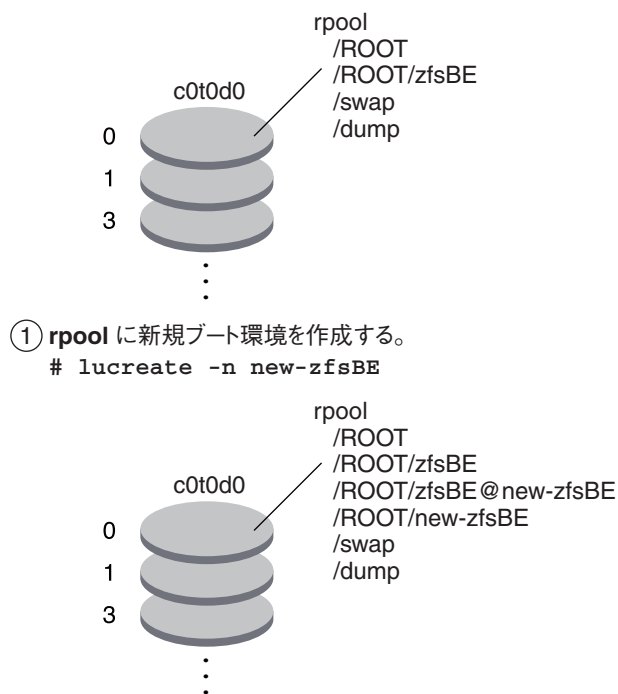
現在のブート環境が同じ ZFS プールにある場合、`-p` オプションは省略します。

次の図に、ZFS ルートプールからの ZFS ブート環境の作成を示します。スライス `c0t0d0s0` に、ZFS ルートプール `rpool` が含まれています。`lucreate` コマンドの `-n` オプションで、作成するブート環境に `new-zfsBE` という名前を割り当てます。元のルートプールのスナップショット `rpool@new-zfsBE` が作成されます。このスナップショットから、新規ブート環境 `new-zfsBE` となるクローンが作成されます。ブート環境 `new-zfsBE` は、すぐにアップグレードおよびアクティブにできます。

図 10-3 同一のルートプール上への新規ブート環境の作成

同一のルートプール上への新規ブート環境の作成

現在稼働中のシステム



例 10-3 同一の ZFS ルートプール内でのブート環境の作成

次の例は、図と同じコマンドを示しています。これは同じルートプール内に新しいブート環境を作成します。`lucreate` コマンドは、`-c zfsBE` オプションで現在稼働中のブート環境を指定し、`-n new-zfsBE` で新しいブート環境を作成します。`zfs list` コマンドでは、新しいブート環境とスナップショットにある ZFS データセットが表示されます。

```
# lucreate -c zfsBE -n new-zfsBE
# zfs list
```

例 10-3 同一の ZFS ルートプール内でのブート環境の作成 (続き)

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool	11.4G	2.95G	31K	/rpool
rpool/ROOT	4.34G	2.95G	31K	legacy
rpool/ROOT/new-zfsBE	4.34G	2.95G	4.34G	/
rpool/dump	2.06G	5.02G	16K	-
rpool/swap	5.04G	7.99G	16K	-

別のルートプール上への新規ブート環境の作成

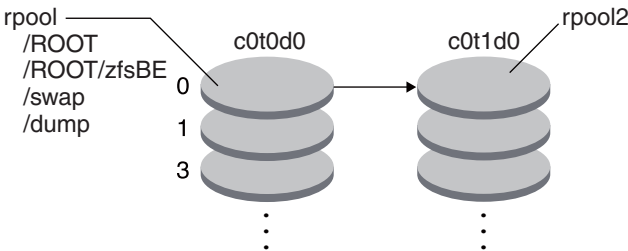
lucreate コマンドを使用して、既存の ZFS ルートプールを別の ZFS ルートプール内にコピーできます。コピー処理には、システムによって時間がかかる場合があります。

次の図に、ブート可能な ZFS ルートプールがまだ存在しないため、ZFS ルートプール rpool2 を c0t1d0s5 上に作成する zpool コマンドを示します。lucreate コマンドの -n オプションで、作成するブート環境に new-zfsBE という名前を割り当てます。-p オプションでは、新しいブート環境を配置する場所を指定します。

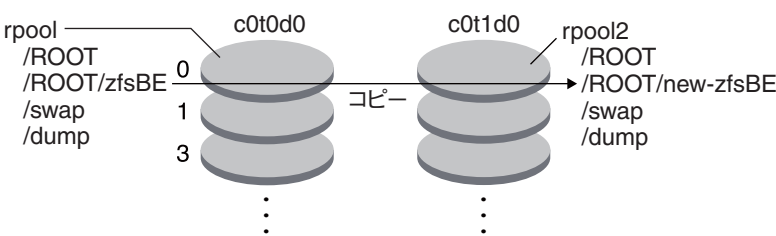
図 10-4 別のルートプール上への新規ブート環境の作成

別のルートプール上への新規ブート環境の作成

- ① 別個のスライス上に新規ルートプールを作成する。
zpool create rpool2 c0t1d0s0



- ② **rpool2** 上に新規ブート環境を作成する。
lucreate -n new-zfsBE -p rpool2



例 10-4 異なる ZFS ルートプール上へのブート環境の作成

次の例は、新しいルートプールの作成後、新しく作成したルートプールに新しいブート環境を作成する、図中のコマンドと同じコマンドを示しています。この例の **zpool create** コマンドで **rpool2** が作成されます。 **zfs list** コマンドで、 **rpool2** に ZFS データセットが作成されていないことが示されます。データセットは、 **lucreate** コマンドで作成されます。

```
# zpool create rpool2 c0t2d0s5
# zfs list
NAME                                USED  AVAIL  REFER  MOUNTPOINT
rpool                               11.4G  2.95G   31K    /rpool
rpool/ROOT                          4.34G  2.95G   31K    legacy
rpool/ROOT/new-zfsBE                4.34G  2.95G  4.34G    /
rpool/dump                          2.06G  5.02G   16K    -
rpool/swap                          5.04G  7.99G   16K    -
```

新しい ZFS ルートプール **rpool2** が、ディスクスライス **c0t2d0s5** 上に作成されます。

```
# lucreate -n new-zfsBE -p rpool2
# zfs list
NAME                                USED  AVAIL  REFER  MOUNTPOINT
```

例 10-4 異なる ZFS ルートプール上へのブート環境の作成 (続き)

rpool	11.4G	2.95G	31K	/rpool
rpool/ROOT	4.34G	2.95G	31K	legacy
rpool/ROOT/new-zfsBE	4.34G	2.95G	4.34G	/
rpool/dump	2.06G	5.02G	16K	-
rpool/swap	5.04G	7.99G	16K	-

新しいブート環境 `new-zfsBE` が、`ROOT`、`dump`、および `swap` の他のデータセットと共に `rpool2` に作成されます。ブート環境 `new-zfsBE` は、すぐにアップグレードおよびアクティブにできます。

現在稼働中のシステム以外のソースからの新規ブート環境の作成

現在稼働中のシステム以外のソースからブート環境を作成する場合は、`lucreate` コマンドに `-s` オプションを指定して実行します。`-s` オプションは、UFS ファイルシステムの場合と同じように機能します。`-s` オプションには、代替ルート (`/`) ファイルシステムのパスを指定します。この代替ルート (`/`) ファイルシステムが、新しい ZFS ルートプールを作成するためのソースとなります。代替ルートは、UFS (`/`) ルートファイルシステム、ZFS ルートプールのどちらでもかまいません。コピー処理には、システムによって時間がかかる場合があります。

例 10-5 代替ルート (`/`) ファイルシステムからのブート環境の作成

次のコマンドで、既存の ZFS ルートプールから新規 ZFS ルートプールを作成します。`-n` オプションで、作成するブート環境に `new-zfsBE` という名前を割り当てます。`-s` オプションには、現在稼働中のブート環境の代わりにコピー元として使用する、ブート環境 `source-zfsBE` を指定します。`-p` オプションでは、新しいブート環境を `newpool2` に配置することを指定します。

```
# lucreate -n new-zfsBE -s source-zfsBE -p rpool2
```

ブート環境 `new-zfsBE` は、すぐにアップグレードおよびアクティブにできます。

非大域ゾーンがインストールされているシステム上への ZFS ブート環境の作成

Live Upgrade を使用して、非大域ゾーンを ZFS ルートファイルシステムに移行できます。概要、計画、および手順については、[第 13 章「非大域ゾーンがインストールされている ZFS での Live Upgrade」](#) を参照してください。

ZFS の Live Upgrade (計画)

この章では、UFS ファイルシステムの ZFS ファイルシステムへの移行を実行する前、または既存の ZFS ルートプールから新規 ZFS ブート環境を作成する前にチェックするガイドラインと要件について説明します。

注 - Oracle Solaris の機能である Live Upgrade を使用したブート環境の作成は、Solaris 10 10/08 リリースの新機能です。UFS ファイルシステムに対して Live Upgrade を実行するときは、コマンド行パラメータにも Live Upgrade の操作にも変更はありません。UFS ファイルシステムがあるシステムで Live Upgrade を実行する場合は、このドキュメントの [パート I 「Live Upgrade を使ったアップグレード」](#) を参照してください。

Live Upgrade 使用時のシステム要件と制限事項

UFS ファイルシステムの ZFS ファイルシステムへの移行を実行する前、または既存の ZFS ルートプールから新規 ZFS ブート環境を作成する前に、必ず、次の要件と制限事項をよく読んで理解してください。これらの要件は、『[Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画](#)』の第 5 章「ZFS ルートファイルシステムのインストール計画」に列挙されている要件に追加されるものです。

表 11-1 要件および制限事項

要件または制限事項	説明	情報
Solaris 10 10/08 以降のリリースをインストールしてください。	Live Upgrade を使った UFS ファイルシステムから ZFS ルートプールへの移行、およびルートプール内への新しいブート環境の作成は、Solaris 10 10/08 リリースの新機能です。このリリースには、ZFS での Live Upgrade の使用に必要なソフトウェアが含まれています。ZFS を使用する場合は、このリリース以降をインストールしてください。	
ディスク容量	ブート可能な ZFS ルートファイルシステムに使用されるプール領域の最小ディスク容量は、物理メモリー容量、使用可能ディスク領域、作成するブート環境の数によって異なります。	詳細については、『 Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画 』の「ZFS インストールのディスク容量要件」を参照してください。

表 11-1 要件および制限事項 (続き)

要件または制限事項	説明	情報
UFS ルート (/) ファイルシステムから ZFS ルートプールに移行する場合は、次の要件を考慮します。	<ul style="list-style-type: none"> ■ UFS ファイルシステムから ZFS ファイルシステムへの移行のみが可能です。 ■ UFS ファイルシステム以外のファイルシステムを ZFS ルートプールに移行することはできません。 ■ UFS ファイルシステムを ZFS ルートプールから作成することはできません。 ■ 移行する前に、ZFS ストレージプールが存在することを確認してください。 ■ ZFS ストレージプールをアップグレード可能かつブート可能にするには、ディスク全体ではなくスライスとして作成します。 <ul style="list-style-type: none"> ■ スライスで作成したプールはミラー化できませんが、複数ディスクの RAID-Z または非冗長構成はできません。SVM デバイス情報を /dev/md/[r]dsk ディレクトリで利用可能にしておいてください。 ■ プールには、SMI ラベルを付けます。EFI ラベルの付いたディスクはブートできません。 ■ x86 のみ: ZFS プールは fdisk パーティションを持つスライス内に作成します。 	<ul style="list-style-type: none"> ■ 詳細な手順については、195 ページの「UFS ファイルシステムの ZFS ファイルシステムへの移行」を参照してください。 ■ SVM の詳細については、『Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画』の「Solaris ボリュームマネージャコンポーネントの概要」を参照してください。 ■ fdisk パーティションについては、『Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画』の「パーティション分割に関する推奨事項」を参照してください。 ■ EFI ラベルの付いたディスクのその他の制約については、『Oracle Solaris の管理: デバイスとファイルシステム』の「ディスク管理の概要」を参照してください。

表 11-1 要件および制限事項 (続き)

要件または制限事項	説明	情報
共有ファイルシステムを移行する場合、共有ファイルシステムを、新しい ZFS ルートプール上の別個のスライスにコピーすることはできません。	たとえば、UFS ルート (/) ファイルシステムで Live Upgrade を実行する場合は、-m オプションを使用して /export ファイルシステムを別のデバイスにコピーできます。共有ファイルシステムを ZFS プールにコピーする -m オプションはありません。	
非大域ゾーンを含む UFS ルートファイルシステムを移行する場合、共有ファイルシステムは移行されません。	UFS ルート (/) ファイルシステムおよび非大域ゾーンがインストールされているシステムでは、非大域ゾーンは、そのゾーンがクリティカルファイルシステム内にある場合に UFS から ZFS への移行の一部として移行されるか、同じ ZFS プール内でアップグレードを行うときにクローニングされます。非大域ゾーンが共有 UFS (/) ファイルシステムに存在する場合、ZFS ルートプールに移行するには、以前の Oracle Solaris リリースの場合と同様に、まずゾーンをアップグレードする必要があります。	<ul style="list-style-type: none">■ クリティカルファイルシステムと共有ファイルシステムについては、22 ページの「ファイルシステムのタイプ」を参照してください。■ 非大域ゾーンがインストールされている場合の手順については、第 13 章「非大域ゾーンがインストールされている ZFS での Live Upgrade」を参照してください。
ZFS rename コマンドは使用しないでください。	Live Upgrade 機能で名前の変更が認識されないため、ludelete などの後続のコマンドが失敗します。既存のブート環境を引き続き使用するのであれば、ZFS のプールやファイルシステムの名前は変更しないでください。	
lucreate コマンドを使用する前に、データセットのプロパティを設定します。	Live Upgrade は、ブート環境のデータセット、およびスワップ領域とダンプデバイスのための ZFS ボリュームを作成しますが、既存のデータセットプロパティを変更することはありません。そのため、新しいブート環境で有効にするデータセットプロパティがある場合は、lucreate 操作の前にそのプロパティを設定する必要があります。例:	『Oracle Solaris ZFS 管理ガイド』の「ZFS のプロパティの概要」 を参照してください。
	<pre># zfs set compression=on rpool/ROOT</pre>	

表 11-1 要件および制限事項 (続き)

要件または制限事項	説明	情報
同一の ZFS ルートプール内に ZFS ブート環境を作成する場合、lucreate コマンドの追加用オプションや除外用オプションを使ってその内容をカスタマイズすることはできません。	<p>同一の ZFS ルートプールにブート環境を作成する場合、プライマリブート環境でファイルを追加したり除外したりするために -f、-o、-y、-Y、および -z オプションを使用することはできません。ただし、次の場合はこれらのオプションを使用できます。</p> <ul style="list-style-type: none">■ UFS ファイルシステムから UFS ファイルシステムにブート環境を作成する場合■ UFS ファイルシステムから ZFS ルートプールにブート環境を作成する場合■ ZFS ルートプールから別の ZFS ルートプールにブート環境を作成する場合	追加用オプションと除外用オプションの使用の詳細については、 88 ページの「ブート環境の作成とコンテンツのカスタマイズ」 を参照してください。

注 - Live Upgrade を使用して、ルート以外の ZFS ファイルシステムをアップグレードすることはできません。

ZFS ルートプールのブート環境の作成

この章では、Live Upgrade を使用して ZFS ブート環境を作成する方法の手順を説明します。

注-Live Upgrade を使った、UFS ファイルシステムから ZFS ルートプールへの移行や ZFS ブート環境の作成は、Solaris 10 10/08 リリースの新機能です。UFS ファイルシステムがあるシステムで Live Upgrade を実行する方法については、このドキュメントのパート I 「[Live Upgrade を使ったアップグレード](#)」を参照してください。

この章で手順を説明するタスクは次のとおりです。

- 195 ページの「UFS ファイルシステムの ZFS ファイルシステムへの移行」
- 202 ページの「同一の ZFS ルートプール内でのブート環境の作成」
- 207 ページの「新しいルートプール内でのブート環境の作成」
- 212 ページの「現在稼働中のシステム以外のソースからのブート環境の作成」
- 213 ページの「ZFS ブート環境へのフォールバック」

非大域ゾーンがインストールされている場合の ZFS の使用手順については、[第 13 章「非大域ゾーンがインストールされている ZFS での Live Upgrade」](#)を参照してください。

UFS ファイルシステムの ZFS ファイルシステムへの移行

ここでは、UFS ファイルシステムを ZFS ファイルシステムに移行する手順について説明します。ブート環境を作成すると、クリティカルファイルシステムをアクティブな UFS ブート環境から ZFS ルートプールにコピーできるようになります。lucreate コマンドで、クリティカルファイルシステムを既存の ZFS ルートプール内の新しいブート環境にコピーします。ユーザー定義の (共有可能な) ファイルシステムはコピーされず、ソース UFS ブート環境と共有されません。また、/swap は、UFS ファイルシステムと ZFS ルートプール間で共有されません。クリティカル

ファイルシステムと共有可能ファイルシステムの概要については、[22 ページ](#)の「[ファイルシステムのタイプ](#)」を参照してください。

▼ UFS ファイルシステムを ZFS ファイルシステムに移行する方法

注 - アクティブな UFS ルート (/) ファイルシステムを ZFS ルートプールに移行するには、ルートプールの名前を指定します。クリティカルファイルシステムは、このルートプールにコピーされます。

- 1 **Live Upgrade** をはじめて実行する前に、インストールメディアから最新の **Live Upgrade** パッケージをインストールし、**My Oracle Support** ナレッジドキュメント **1004881.1 - Live Upgrade Software Patch Requirements** (以前の **206844**) に記載されているパッチをインストールする必要があります。[My Oracle Support Web](#) サイトでナレッジドキュメント **1004881.1 - Live Upgrade Software Patch Requirements** (以前の **206844**) を検索してください。

最新のパッケージとパッチにより、リリースに最新のバグ修正と新機能がすべて含まれるようになります。新しいブート環境の作成に進む前に、システムに関連するすべてのパッチを必ずインストールしてください。

次のサブステップは、**My Oracle Support** ナレッジドキュメント **1004881.1 - Live Upgrade Software Patch Requirements** (以前の **206844**) の手順を説明したものです。

注 - **Live Upgrade** を使って新規 ZFS ブート環境を作成するためには、少なくとも **Solaris 10 10/08** リリースがインストールされている必要があります。それより前のリリースの ZFS および **Live Upgrade** ソフトウェアでは、このタスクを実行することはできません。

- a. スーパーユーザーになるか、同等の役割を引き受けます。

注 - 役割には、認証と特権コマンドが含まれます。役割の詳細については、『**Solaris のシステム管理: セキュリティサービス**』の「**RBAC の構成 (タスクマップ)**」を参照してください。

- b. **My Oracle Support Web** サイトで、ナレッジドキュメント **1004881.1 - Live Upgrade Software Patch Requirements** (以前の **206844**) の手順に従って、**Live Upgrade** パッケージを削除して追加します。

3 つの **Live Upgrade** パッケージ **SUNWluu**、**SUNWlur**、および **SUNWlucfg** は、**Live Upgrade** を使ってアップグレードするために必要なソフトウェアを構成します。これらのパッケージには、既存のソフトウェア、新しい機能、およびバグ修

正が含まれています。Live Upgrade を使用する前に、既存パッケージを削除しないで新しいパッケージをシステムにインストールすると、ターゲットリリースへのアップグレードは失敗します。SUMWLucfg パッケージは、**Solaris 10 8/07** 以降のリリースの新機能です。Solaris 10 8/07 より前のリリースから Live Upgrade パッケージを使用する場合は、このパッケージを削除する必要はありません。

```
# pkgrm SUNWLucfg SUNWluu SUNWLur
```

- c. アップグレードのターゲットリリースの新しい **Live Upgrade** パッケージをインストールします。手順は、[62 ページの「Live Upgrade のインストール」](#)を参照してください。
- d. **Live Upgrade** を実行する前に、次のパッチをインストールする必要があります。これらのパッチにより、リリースに最新のバグ修正と新機能がすべて含まれるようになります。

[My Oracle Support](#) を調べて最新の更新済みパッチリストを持っていることを確認してください。My Oracle Support Web サイトでナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) を検索してください。

- パッチをローカルディスクに保存する場合は、`/var/tmp/lupatches` のようなディレクトリを作成し、パッチをそのディレクトリにダウンロードします。
- [My Oracle Support](#) Web サイトから、パッチの一覧を取得します。
- パッチのディレクトリに移動します。

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

- `patchadd` コマンドを使用してパッチをインストールします。

```
# patchadd patch_id
```

`patch_id` はパッチの番号です。複数のパッチ名を指定する場合は、スペースで区切ります。

注 - パッチはナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) に指定されている順序で適用する必要があります。

- 必要に応じてシステムをリブートします。いくつかのパッチは、有効にするためにリブートする必要があります。

x86 のみ: システムをリブートする必要があります。そうしないと、Live Upgrade は失敗します。

```
# init 6
```

以上で正しく移行するために必要なパッケージとパッチがインストールされました。

2 ZFS ルートプールを作成します。

ZFS ルートプールは、ブート可能かつアップグレード可能にするため単一のスライス上に作成します。

zpool create rpool c0t1d0s5

rpool 作成する新しい ZFS ルートプールの名前を指定します。

c0t1d0s5 新しいルートプールをディスクスライス *c0t1d0s5* 上に作成します。

新規ルートプールの作成の詳細については、『[Oracle Solaris ZFS 管理ガイド](#)』を参照してください。

3 UFS ルート (/) ファイルシステムを新しい ZFS ルートプールに移行します。

lucreate [-c ufsBE] -n new-zfsBE -p rpool

ufsBE 現在の UFS ブート環境の名前。このオプションは必須ではなく、また、最初のブート環境の作成時にのみ使用します。lucreate コマンドを初めて実行する場合に -c オプションを省略すると、デフォルトの名前が自動的に作成されます。

new-zfsBE 作成されるブート環境の名前。名前は、システム上で一意となるように指定する必要があります。

-p rpool 新しく作成した ZFS ルート (/) ファイルシステムを、*rpool* で定義した ZFS ルートプールに配置します。

新しい ZFS ブート環境の作成には、しばらく時間がかかることがあります。UFS ファイルシステムのデータが ZFS ルートプールにコピーされます。非アクティブなブート環境が作成された場合は、luupgrade コマンドまたは luactivate コマンドを使用して、新しい ZFS ブート環境をアップグレードまたはアクティブにすることができます。

4 (オプション) ブート環境が完成したことを確認します。

# 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 (オプション) システム上の基本データセット情報を確認します。

list コマンドで、システム上のすべてのデータセットの名前を表示できます。この例では、rpool が ZFS プールの名前、new-zfsBE が新しく作成した ZFS ブート環境の名前です。

# zfs list					
NAME	USED	AVAIL	REFER	MOUNTPOINT	
rpool	11.4G	2.95G	31K	/rpool	
rpool/ROOT	4.34G	2.95G	31K	legacy	
rpool/ROOT/new-zfsBE	4.34G	2.95G	4.34G	/	

```

rpool/dump                2.06G  5.02G   16K  -
rpool/swap                5.04G  7.99G   16K  -

```

表示される新しいブート環境のマウントポイントは、luactivate コマンドが実行されるまでの一時的なものです。/dump ボリュームと /swap ボリュームは、元の UFS ブート環境と共有されませんが、ZFS ルートプール内およびルートプール内のブート環境内で共有されます。

これで、新しいブート環境をアップグレードおよびアクティブにできます。

例 12-1 UFS ルート (/) ファイルシステムの ZFS ルートプールへの移行

この例では、新しい ZFS ルートプール rpool が、別のスライス c0t0d0s4 上に作成されます。lucreate コマンドが、現在稼働中の UFS ブート環境 c0t0d0 を新しい ZFS ブート環境 new-zfsBE に移行し、その新しいブート環境を rpool に配置します。

```

# 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
Determining types of file systems supported
Validating file system requests
Preparing logical storage devices
Preparing physical storage devices
Configuring physical storage devices
Configuring logical storage devices
Analyzing system configuration.
No name for current boot environment.
Current boot environment is named <c0t0d0>.
Creating initial configuration for primary boot environment <c0t0d0>.
INFORMATION: No BEs are configured on this system.
The device </dev/dsk/clt0d0s0> is not a root device for any boot
environment; cannot get BE ID.
PBE configuration successful: PBE name <c0t0d0> PBE Boot Device
</dev/dsk/clt0d0s0>.
Updating boot environment description database on all BEs.
Updating system configuration files.
The device </dev/dsk/clt0d0s6> 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 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>.
Analyzing zones.
Mounting ABE <new-zfsBE>.
Generating file list.
Copying data from PBE <c0t0d0> to ABE <new-zfsBE>.
100% of filenames transferred
Finalizing ABE.
Fixing zonepaths in ABE.

```

```
Unmounting ABE <new-zfsBE>.
Fixing properties on ZFS datasets in ABE.
Reverting state of zones in PBE <c0t0d0>.
Making boot environment <new-zfsBE> bootable.
Creating boot_archive for /.alt.tmp.b-Cjh.mnt
updating /.alt.tmp.b-Cjh.mnt/platform/sun4u/boot_archive
Population of boot environment <new-zfsBE> successful.
Creation of boot environment <new-zfsBE> successful.
```

```
# lustatus
boot environment      Is      Active      Active      Can      Copy
Name                 Complete Now      OnReboot    Delete    Status
-----
c0t0d0               yes      yes      yes         no        -
new-zfsBE             yes      no       no         yes       -

# zfs list
NAME                                USED  AVAIL  REFER  MOUNTPOINT
rpool                             11.4G  2.95G   31K    /rpool
rpool/ROOT                     4.34G  2.95G   31K    legacy
rpool/ROOT/new-zfsBE           4.34G  2.95G  4.34G    /
rpool/dump                     2.06G  5.02G   16K    -
rpool/swap                     5.04G  7.99G   16K    -
```

これで、新しいブート環境をアップグレードまたはアクティブにできます。

この例では、luupgrade コマンドを使って、-s オプションで指定された場所に格納されているイメージから新しいブート環境をアップグレードしています。

```
# 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.

新しいブート環境は、作成後いつでもアクティブにできます。

luactivate new-zfsBE

A Live Upgrade Sync operation will be performed on startup of boot
 environment <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. Boot the machine to Single User mode using a different boot device
 (like the Solaris Install CD or Network). Examples:

 At the PROM monitor (ok prompt):
 For boot to Solaris CD: boot cdrom -s
 For boot to network: boot net -s
3. Mount the Current boot environment root slice to some directory (like
 /mnt). You can use the following command to mount:

 mount -Fufs /dev/dsk/clt0d0s0 /mnt
4. Run <luactivate> utility with out any arguments from the current boot
 environment root slice, as shown below:

 /mnt/sbin/luactivate
5. luactivate, activates the previous working boot environment and
 indicates the result.
6. Exit Single User mode and reboot the machine.

Modifying boot archive service
 Activation of boot environment <new-zfsBE> successful.

システムを ZFS ブート環境にリブートします。

```
# init 6
# svc.startd: The system is coming down. Please wait.
svc.startd: 79 system services are now being stopped.
.
.
.
```

UFS ブート環境に戻す場合は、ZFS ブート環境で作成したすべての ZFS ストレージプールをもう一度インポートしてください。これらは、UFS ブート環境で自動的に使用可能にはならないからです。元の UFS ブート環境に切り替えると、次の例のようなメッセージが表示されます。

```
# 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>
```

同一の ZFS ルートプール内でのブート環境の作成

次の手順で、既存の ZFS ルートプール内に新しい ZFS ブート環境を作成する方法について説明します。非アクティブなブート環境の作成後、新しいブート環境を都合のよいときにアップグレードおよびアクティブにできます。同一プール内でブート環境を作成する場合、-p オプションは不要です。

▼ ZFS ブート環境を同一の ZFS ルートプール内に作成する方法

- 1 **Live Upgrade** をはじめて実行する前に、インストールメディアから最新の **Live Upgrade** パッケージをインストールし、**My Oracle Support** ナレッジドキュメント **1004881.1 – Live Upgrade Software Patch Requirements** (以前の **206844**) に記載されているパッチをインストールする必要があります。[My Oracle Support Web](#) サイトでナレッジドキュメント **1004881.1 - Live Upgrade Software Patch Requirements** (以前の **206844**) を検索してください。

最新のパッケージとパッチにより、リリースに最新のバグ修正と新機能がすべて含まれるようになります。新しいブート環境の作成に進む前に、システムに関連するすべてのパッチを必ずインストールしてください。

次のサブステップは、My Oracle Support ナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) の手順を説明したものです。

注 - Live Upgrade を使って新規 ZFS ブート環境を作成するためには、少なくとも Solaris 10 10/08 リリースがインストールされている必要があります。それより前のリリースの ZFS および Live Upgrade ソフトウェアでは、このタスクを実行することはできません。

- a. スーパーユーザーになるか、同等の役割を引き受けます。

注 - 役割には、認証と特権コマンドが含まれます。役割の詳細については、『[Solaris のシステム管理: セキュリティーサービス](#)』の「[RBAC の構成 \(タスクマップ\)](#)」を参照してください。

- b. **My Oracle Support Web** サイトで、ナレッジドキュメント **1004881.1 - Live Upgrade Software Patch Requirements** (以前の **206844**) の手順に従って、**Live Upgrade** パッケージを削除して追加します。

3 つの Live Upgrade パッケージ **SUNWluu**、**SUNWlur**、および **SUNWlucfg** は、Live Upgrade を使ってアップグレードするために必要なソフトウェアを構成します。これらのパッケージには、既存のソフトウェア、新しい機能、およびバグ修正が含まれています。Live Upgrade を使用する前に、既存パッケージを削除しないで新しいパッケージをシステムにインストールすると、ターゲットリリースへのアップグレードは失敗します。**SUNWlucfg** パッケージは、**Solaris 10 8/07** 以降のリリースの新機能です。Solaris 10 8/07 より前のリリースから Live Upgrade パッケージを使用する場合は、このパッケージを削除する必要はありません。

注 - **SUNWlucfg** パッケージは、**Solaris 10 8/07** 以降のリリースの新機能です。それより前のリリースの Live Upgrade パッケージを使用する場合は、このパッケージを削除する必要はありません。

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- c. 新しい **Live Upgrade** パッケージをインストールします。手順は、[62 ページの「Live Upgrade のインストール」](#)を参照してください。
- d. **Live Upgrade** を実行する前に、次のパッチをインストールする必要があります。これらのパッチにより、リリースに最新のバグ修正と新機能がすべて含まれるようになります。

[My Oracle Support](#) を調べて最新の更新済みパッチリストを持っていることを確認してください。My Oracle Support Web サイトでナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) を検索してください。

- パッチをローカルディスクに保存する場合は、`/var/tmp/lupatches` のようなディレクトリを作成し、パッチをそのディレクトリにダウンロードします。
- [My Oracle Support](#) Web サイトから、パッチの一覧を取得します。
- この例のようにパッチディレクトリに移動します。

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

- `patchadd` コマンドを使用してパッチをインストールします。

```
# patchadd -M path-to-patches patch_id patch_id
```

`path-to-patches` は、`/var/tmp/lupatches` などのパッチディレクトリへのパスです。`patch_id` はパッチの番号です。複数のパッチ名を指定する場合は、スペースで区切ります。

注-パッチはナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) に指定されている順序で適用する必要があります。

- 必要に応じてシステムをリブートします。いくつかのパッチは、有効にするためにリブートする必要があります。

x86 のみ: システムをリブートする必要があります。そうしないと、Live Upgrade は失敗します。

```
# init 6
```

以上で新しいブート環境を正しく作成するために必要なパッケージとパッチがインストールされました。

2 新しいブート環境を作成します。

```
# lucreate [-c zfsBE] -n new-zfsBE
```

zfsBE 現在のブート環境の名前。このオプションは必須ではなく、また、最初のブート環境の作成時にのみ使用します。`lucreate` をはじめて実行する場合に `-c` オプションを省略すると、ソフトウェアによりデフォルトの名前が作成されます。

new-zfsBE 作成されるブート環境の名前。名前は、システム上で一意となるように指定する必要があります。

新しいブート環境の作成は、きわめて短時間で完了します。現在の ZFS ルートプール内の各データセットのスナップショットが作成され、各スナップショットからクローンが作成されます。スナップショットはディスク容量の効率が非常に高いので、この処理は最小限のディスク容量しか消費しません。非アクティブなブート

環境が作成された場合は、`luupgrade` コマンドまたは `luactivate` コマンドを使用して、新しい ZFS ブート環境をアップグレードまたはアクティブにすることができます。

3 (オプション) ブート環境が完成したことを確認します。

`lustatus` コマンドは、ブート環境の作成が完了してブート可能であるかどうかを報告します。

```
# 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 (オプション) システム上の基本データセット情報を確認します。

この例では、ZFS ルートプールの名前は `rpool` で、`@` 記号はスナップショットを示します。新しいブート環境のマウントポイントは、`luactivate` コマンドが実行されるまでの一時的なものです。`/dump` ボリュームと `/swap` ボリュームは、ZFS ルートプールおよびルートプール内のブート環境と共有されます。

```
# zfs list
NAME                                     USED  AVAIL  REFER  MOUNTPOINT
rpool                                  11.4G  2.95G   31K    /rpool
rpool/ROOT                             4.34G  2.95G   31K    legacy
rpool/ROOT/new-zfsBE                   4.34G  2.95G  4.34G    /
rpool/dump                             2.06G  5.02G   16K    -
rpool/swap                             5.04G  7.99G   16K    -
```

これで、新しいブート環境をアップグレードおよびアクティブにできます。

例 12-2 同一の ZFS ルートプール内でのブート環境の作成

この例では新しい ZFS ブート環境 `new-zfsBE` が作成されます。このブート環境は同一のルートプール内に作成されるため、`-p` オプションは不要です。

```
# 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  Is      Active  Active  Can      Copy
```

Name	Complete	Now	OnReboot	Delete	Status
zfsBE	yes	yes	yes	no	-
new-zfsBE	yes	no	no	yes	-
# zfs list					
NAME				USED	AVAIL
rpool				11.4G	2.95G
rpool/ROOT				4.34G	2.95G
rpool/ROOT/new-zfsBE				4.34G	2.95G
rpool/dump				2.06G	5.02G
rpool/swap				5.04G	7.99G

これで、新しいブート環境をアップグレードおよびアクティブにできます。ZFSブート環境のアップグレードの例については、[例 12-1](#) を参照してください。luupgrade コマンドのその他の使用例については、[第 5 章「Live Upgrade によるアップグレード \(タスク\)」](#) を参照してください。

```
# luactivate new-zfsBE
A Live Upgrade Sync operation will be performed on startup of boot
environment <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. Boot the machine to Single User mode using a different boot device
(like the Solaris Install CD or Network). Examples:

    At the PROM monitor (ok prompt):
    For boot to Solaris CD:  boot cdrom -s
    For boot to network:    boot net -s

3. Mount the Current boot environment root slice to some directory (like
/mnt). You can use the following command to mount:

    mount -Fufs /dev/dsk/clt0d0s0 /mnt

4. Run <luactivate> utility with out any arguments from the current boot
environment root slice, as shown below:

    /mnt/sbin/luactivate

5. luactivate, activates the previous working boot environment and
indicates the result.
```

6. Exit Single User mode and reboot the machine.

Modifying boot archive service
Activation of boot environment <new-zfsBE> successful.

システムを ZFS ブート環境にリブートします。

```
# init 6
# svc.startd: The system is coming down. Please wait.
svc.startd: 79 system services are now being stopped.
.
.
.
```

新しいルートプール内でのブート環境の作成

ここでは、既存の ZFS ルートプールがある場合に、新規のルートプール内に新規の ZFS ブート環境を作成する手順について説明します。非アクティブなブート環境の作成後、新しいブート環境を都合のよいときにアップグレードおよびアクティブにできます。新しいブート環境を配置する場所を指定する `-p` オプションは必須です。既存の ZFS ルートプールは、ブート可能かつアップグレード可能にするため単一のスライス上に配置してください。

▼ ブート環境を新規 ZFS ルートプールに作成する方法

- 1 **Live Upgrade** をはじめて実行する前に、インストールメディアに含まれている最新の **Live Upgrade** パッケージと、ナレッジドキュメントに記載されているパッチをインストールする必要があります。[My Oracle Support Web](#) サイトでナレッジドキュメント **1004881.1 - Live Upgrade Software Patch Requirements** (以前の **206844**) を検索してください。

最新のパッケージとパッチにより、リリースに最新のバグ修正と新機能がすべて含まれるようになります。新しいブート環境の作成に進む前に、システムに関連するすべてのパッチを必ずインストールしてください。

次のサブステップは、My Oracle Support のナレッジドキュメント **1004881.1 - Live Upgrade Software Patch Requirements** (以前の **206844**) の手順を説明したものです。

注 - Live Upgrade を使って新規 ZFS ブート環境を作成するためには、少なくとも Solaris 10 10/08 リリースがインストールされている必要があります。それより前のリリースの ZFS および Live Upgrade ソフトウェアでは、このタスクを実行することはできません。

- a. スーパーユーザーになるか、同等の役割を引き受けます。

注 - 役割には、認証と特権コマンドが含まれます。役割の詳細については、『[Solaris のシステム管理: セキュリティーサービス](#)』の「[RBAC の構成 \(タスクマップ\)](#)」を参照してください。

- b. **My Oracle Support Web** サイトで、ナレッジドキュメント **1004881.1** の手順に従って、**Live Upgrade** パッケージを削除して追加します。

3 つの Live Upgrade パッケージ SUNWluu、SUNWlur、および SUNWlucfg は、Live Upgrade を使ってアップグレードするために必要なソフトウェアを構成します。これらのパッケージには、既存のソフトウェア、新しい機能、およびバグ修正が含まれています。Live Upgrade を使用する前に、既存パッケージを削除しないで新しいパッケージをシステムにインストールすると、ターゲットリリースへのアップグレードは失敗します。SUNWlucfg パッケージは、**Solaris 10 8/07** 以降のリリースの新機能です。Solaris 10 8/07 より前のリリースから Live Upgrade パッケージを使用する場合は、このパッケージを削除する必要はありません。

注 - SUNWlucfg パッケージは、**Solaris 10 8/07** 以降のリリースの新機能です。それより前のリリースの Live Upgrade パッケージを使用する場合は、このパッケージを削除する必要はありません。

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- c. 新しい **Live Upgrade** パッケージをインストールします。手順は、[62 ページの「Live Upgrade のインストール」](#)を参照してください。
- d. **Live Upgrade** を実行する前に、次のパッチをインストールする必要があります。これらのパッチにより、リリースに最新のバグ修正と新機能がすべて含まれるようになります。

[My Oracle Support](#) を調べて最新の更新済みパッチリストを持っていることを確認してください。My Oracle Support で、ナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) を検索してください。

- パッチをローカルディスクに保存する場合は、`/var/tmp/lupatches` のようなディレクトリを作成し、パッチをそのディレクトリにダウンロードします。

- [My Oracle Support](#) Web サイトから、パッチの一覧を取得します。
- この例のようにパッチディレクトリに移動します。

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

- patchadd コマンドを使用してパッチをインストールします。

```
# patchadd -M path-to-patches patch_id patch_id
```

path-to-patches は、*/var/tmp/lupatches* などのパッチディレクトリへのパスです。*patch_id* はパッチの番号です。複数のパッチ名を指定する場合は、スペースで区切ります。

注 - パッチは、My Oracle Support のナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) に指定されている順序で適用する必要があります。

- 必要に応じてシステムをリブートします。いくつかのパッチは、有効にするためにリブートする必要があります。

x86 のみ: システムをリブートする必要があります。そうしないと、Live Upgrade は失敗します。

```
# init 6
```

以上で正しく移行するために必要なパッケージとパッチがインストールされました。

2 ZFS ルートプールを作成します。

ZFS ルートプールは、ブート可能かつアップグレード可能にするため単一のスライス上に作成します。

```
# zpool create rpool2 c0t1d0s5
```

rpool2 新しい ZFS ルートプールの名前です。

c0t1d0s5 *rpool2* を、ブート可能なスライス *c0t1d0s5* に配置するように指定します。

新規ルートプールの作成の詳細については、『[Oracle Solaris ZFS 管理ガイド](#)』を参照してください。

3 新しいブート環境を作成します。

```
# lucreate [-c zfsBE] -n new-zfsBE -p rpool2
```

zfsBE 現在の ZFS ブート環境の名前。

new-zfsBE 作成されるブート環境の名前。名前は、システム上で一意となるように指定する必要があります。

-p *rpool2* 新しく作成した ZFS ルートのブート環境を *rpool2* で定義した ZFS ルートプールに配置します。

新しい ZFS ブート環境の作成には、しばらく時間がかかることがあります。ファイルシステムのデータが新しい ZFS ルートプールにコピーされます。非アクティブなブート環境が作成された場合は、`luupgrade` コマンドまたは `luactivate` コマンドを使用して、新しい ZFS ブート環境をアップグレードまたはアクティブにすることができます。

4 (オプション) ブート環境が完成したことを確認します。

`lustatus` コマンドは、ブート環境の作成が完了してブート可能であるかどうかを報告します。

```
# 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 (オプション) システム上の基本データセット情報を確認します。

次の例では、システム上のすべてのデータセットの名前を表示しています。表示される新しいブート環境のマウントポイントは、`luactivate` コマンドが実行されるまでの一時的なものです。新しいブート環境は、ボリューム `rpool2/dump` および `rpool2/swap` を、`rpool2` ZFS ブート環境と共有します。

```
# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool	11.4G	2.95G	31K	/rpool
rpool/ROOT	4.34G	2.95G	31K	legacy
rpool/ROOT/new-zfsBE	4.34G	2.95G	4.34G	/
rpool/dump	2.06G	5.02G	16K	-
rpool/swap	5.04G	7.99G	16K	-

これで、新しいブート環境をアップグレードおよびアクティブにできます。

例 12-3 新しいルートプールでのブート環境の作成

この例では、新しい ZFS ルートプール `newPool` が別のスライス `c0t1s0s5` 上に作成されます。`lucreate` コマンドで、新しい ZFS ブート環境 `new-zfsbe` を作成します。このブート環境は異なるルートプール内に作成されるため、`-p` オプションは必須です。

```
# zpool create newPool C0t2d0s5
```

```
# zfs list
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
newPool	92.5K	18.7G	31K	/newPool
rpool	11.4G	2.95G	31K	/rpool
rpool/ROOT	4.34G	2.95G	31K	legacy
rpool/ROOT/zfsBE	4.34G	2.95G	4.34G	/
rpool/dump	2.06G	5.02G	16K	-
rpool/swap	5.04G	7.99G	16K	-

```
# lucreate -c c0t1d0s5 -n new-zfsbe -p newPool
Checking GRUB menu...
Analyzing system configuration.
Updating boot environment description database on all BEs.
Updating system configuration files.
The device </dev/dsk/clt1d0s0> is not a root device for any boot environment; cannot get BE ID.
Creating configuration for boot environment <new-zfsbe>.
Source boot environment is <c0t1d0s5>.
Creating file systems on boot environment <new-zfsbe>.
Creating <zfs> file system for </> in zone <global> on <newPool/ROOT/new-zfsbe>.
Populating file systems on boot environment <new-zfsbe>.
Analyzing zones.
Mounting ABE <new-zfsbe>.
Generating file list.
Copying data from PBE <c0t1d0s5> to ABE <new-zfsbe>.
100% of filenames transferred
Finalizing ABE.
Fixing zonepaths in ABE.
Unmounting ABE <new-zfsbe>.
Fixing properties on ZFS datasets in ABE.
Reverting state of zones in PBE <c0t1d0s5>.
Making boot environment <new-zfsbe> bootable.
Updating bootenv.rc on ABE <new-zfsbe>.
Saving existing file </boot/grub/menu.lst> in top level dataset for BE <new-zfsbe> as <mount-point>
//boot/grub/menu.lst.prev.
File </boot/grub/menu.lst> propagation successful
Copied GRUB menu from PBE to ABE
No entry for BE <new-zfsbe> in GRUB menu
Population of boot environment <new-zfsbe> successful.
Creation of boot environment <new-zfsbe> successful.
```

lustatus

Boot Environment Name	Is Complete	Active Now	Active On Reboot	Can Delete	Copy Status
c0t0d0	yes	yes	yes	no	-
zfsBE		yes	no	no	yes -
new-zfsbe	yes	no	no	yes	-

zfs list

NAME	USED	AVAIL	REFER	MOUNTPOINT
newPool	7.15G	11.6G	36K	/newPool
newPool/ROOT	4.05G	11.6G	31K	legacy
newPool/ROOT/new-zfsbe	4.05G	11.6G	4.05G	/
newPool/dump	1.03G	12.6G	16K	-
newPool/swap	2.06G	13.6G	16K	-
rpool		11.4G	2.95G	31K /rpool
rpool/ROOT		4.34G	2.95G	31K legacy
rpool/ROOT/zfsBE		4.34G	2.95G	4.34G /
rpool/dump		2.06G	5.02G	16K -
rpool/swap		5.04G	7.99G	16K -

現在稼働中のシステム以外のソースからのブート環境の作成

ここでは、現在アクティブなブート環境として使用されていない既存の ZFS ルートプールまたは UFS ブート環境がある場合に、新しい ZFS ブート環境をこのブート環境から作成する手順について説明します。新しい ZFS ブート環境の作成後、この新しいブート環境を都合のよいときにアップグレードおよびアクティブにできます。

現在稼働中のシステム以外のソースからブート環境を作成する場合は、lucreate コマンドに -s オプションを指定して実行します。-s オプションは、UFS ファイルシステムの場合と同じように機能します。-s オプションには、代替ルート (/) ファイルシステムのパスを指定します。この代替ルート (/) ファイルシステムが、新しい ZFS ルートプールを作成するためのソースとなります。代替ルートは、UFS (/) ルートファイルシステム、ZFS ルートプールのどちらでもかまいません。コピー処理には、システムによって時間がかかる場合があります。

次の例では、別の ZFS ルートプール上にブート環境を作成する場合の、-s オプションの使用方法を示します。

例 12-4 現在稼働中のシステム以外のソースからブート環境を作成する方法

次のコマンドで、既存の ZFS ルートプールから新規 ZFS ルートプールを作成します。-n オプションで、作成するブート環境に new-zfsBE という名前を割り当てます。-s オプションには、現在稼働中のブート環境の代わりにコピー元として使用する、ブート環境 rpool3 を指定します。-p オプションでは、新しいブート環境を rpool2 に配置することを指定します。

```
# lucreate -n new-zfsBE -s rpool3 -p rpool2
# lustatus
boot environment      Is      Active  Active  Can      Copy
Name                  Complete Now    OnReboot Delete   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
rpool                               11.4G  2.95G   31K    /rpool
rpool/ROOT                        4.34G  2.95G   31K    legacy
rpool/ROOT/new-zfsBE              4.34G  2.95G  4.34G    /
rpool/dump                        2.06G  5.02G   16K    -
rpool/swap                        5.04G  7.99G   16K    -
```

これで、新しいブート環境をアップグレードおよびアクティブにできます。

ZFS ブート環境へのフォールバック

アップグレード後にエラーが検出された場合、またはアップグレードしたコンポーネントとアプリケーションの間に互換性がない場合には、`luactivate` コマンドを使って元のブート環境に戻すことができます。

UFS ブート環境から ZFS ルートプールに移行した場合に、UFS ブート環境に戻すには、ZFS ブート環境に作成されたすべての ZFS ストレージプールをもう一度インポートしてください。これらの ZFS ストレージプールは、UFS ブート環境で自動的に使用可能にはなりません。元の UFS ブート環境に切り替えると、次の例のようなメッセージが表示されます。

```
# 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>
```

元のブート環境に戻す例については、[第6章「障害回復: 元のブート環境へのフォールバック \(タスク\)」](#)を参照してください。

非大域ゾーンがインストールされている ZFS での Live Upgrade

この章では、UFS (/) ルートファイルシステムの ZFS ルートプールへの移行に関する概要と詳細な手順について説明します。ここには、次の情報が含まれています。

- [215 ページの「非大域ゾーンがインストールされているシステム上への ZFS ブート環境の作成 \(概要と計画\)」](#)
- [216 ページの「非大域ゾーンがインストールされている UFS ルート \(/\) ファイルシステムから ZFS ルートプールへの移行 \(タスク\)」](#)

注 - Oracle Solaris の機能である Live Upgrade を使用した、UFS ルート (/) ファイルシステムから ZFS ルートプールへの移行や ZFS ブート環境の作成は、Solaris 10 10/08 リリースの新機能です。UFS ファイルシステムに対して Live Upgrade を実行するときは、コマンド行パラメータにも Live Upgrade の操作にも変更はありません。UFS ファイルシステムが存在するシステムで Live Upgrade を実行する場合は、このドキュメントの [パート I 「Live Upgrade を使ったアップグレード」](#) を参照してください。

非大域ゾーンがインストールされているシステム上への ZFS ブート環境の作成 (概要と計画)

Live Upgrade を使用して、非大域ゾーンがインストールされている UFS ルート (/) ファイルシステムを ZFS ルートプール上に移行できます。ファイルシステムに関連するすべての非大域ゾーンも、新しいブート環境にコピーされます。次の表に一覧で示す非大域ゾーンの移行シナリオがサポートされています。

移行前のルートファイルシステムとゾーンの組み合わせ	移行後のルートファイルシステムとゾーンの組み合わせ
UFS ファイルシステム内で非大域ゾーンのルートディレクトリを持つ UFS ルートファイルシステム	UFS ルートファイルシステム内で非大域ゾーンのルートディレクトリを持つ UFS ルートファイルシステム ZFS ルートプール内で非大域ゾーンのルートディレクトリを持つ ZFS ルートプール UFS ファイルシステム内で非大域ゾーンのルートディレクトリを持つ ZFS ルートプール
ZFS ルートプール内で非大域ゾーンのルートを持つ UFS ルートファイルシステム	ZFS ルートプール内で非大域ゾーンのルートを持つ ZFS ルートプール ZFS ルートプール内で非大域ゾーンのルートを持つ UFS ルートファイルシステム
ZFS ルートプール内で非大域ゾーンのルートディレクトリを持つ ZFS ルートプール	ZFS ルートプール内で非大域ゾーンのルートディレクトリを持つ ZFS ルートプール

UFS ルート (/) ファイルシステムおよび非大域ゾーンがインストールされているシステムでは、非大域ゾーンは、そのゾーンが非共有ファイルシステム内にある場合に UFS から ZFS への移行の一部として移行されるか、同じ ZFS プール内でアップグレードを行うときにクローニングされます。非大域ゾーンが共有 UFS ファイルシステムに存在する場合、別の ZFS ルートプールに移行するには、以前の Solaris リリースの場合と同様に、まず非大域ゾーンをアップグレードします。

- ZFS ルートプールへの移行の計画の詳細については、[189 ページの「Live Upgrade 使用時のシステム要件と制限事項」](#)を参照してください。
- ZFS および非大域ゾーンのその他の制限事項については、『[Oracle Solaris ZFS 管理ガイド](#)』の「[ゾーンがインストールされている Solaris システムで ZFS を使用する](#)」を参照してください。

非大域ゾーンがインストールされている **UFS** ルート (/) ファイルシステムから **ZFS** ルートプールへの移行 (タスク)

このセクションでは、非大域ゾーンがインストールされているシステムで UFS ルート (/) ファイルシステムから ZFS ルートプールに移行するための手順を説明します。非大域ゾーンは、UFS ファイルシステムの共有ファイルシステム上にあります。

▼ 非大域ゾーンのあるシステムで **UFS** ファイルシステムを **ZFS** ルートプールに移行する方法

lucreate コマンドは、UFS ルート (/) ファイルシステムから ZFS ルートプールのブート環境を作成します。ZFS ルートプールは lucreate を使用する前に作成します。また、アップグレード可能かつブート可能にするために、ディスク全体ではなくスライス上に作成します。次に示す手順では、UFS ルート (/) ファイルシステムに関連する既存の非大域ゾーンが、ZFS ルートプール内の新しいブート環境にコピーされます。

- 1 はじめて **Live Upgrade** を実行するときには、次の手順を実行します。

注 - Live Upgrade を使用して新しい ZFS ブート環境を作成するためには、少なくとも **Solaris 10 10/08** リリースがインストールされている必要があります。それより前のリリースの ZFS および Live Upgrade ソフトウェアでは、このタスクを実行することはできません。

- a. 必要に応じて、システム上の既存の **Live Upgrade** パッケージを削除します。新しいリリースにアップグレードする場合は、そのリリースのパッケージをインストールしてください。

3 つの Live Upgrade パッケージ SUNWluu、SUNWlur、および SUNWlucfg は、Live Upgrade を使ってアップグレードするために必要なソフトウェアを構成します。これらのパッケージには、既存のソフトウェア、新しい機能、およびバグ修正が含まれています。Live Upgrade を使用する前に、既存パッケージを削除しないで新しいパッケージをシステムにインストールすると、ターゲットリリースへのアップグレードは失敗します。

```
# pkgrm SUNWlucfg SUNWluu SUNWlur
```

- b. アップグレードのターゲットリリースの新しい **Live Upgrade** パッケージをインストールします。手順は、[62 ページの「Live Upgrade のインストール」](#)を参照してください。
- c. **Live Upgrade** をインストールまたは実行する前に、次のパッチをインストールする必要があります。これらのパッチにより、リリースに最新のバグ修正と新機能がすべて含まれるようになります。

[My Oracle Support](#) を調べて最新の更新済みパッチリストを持っていることを確認してください。My Oracle Support で、ナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) を検索してください。

- スーパーユーザーになるか、同等の役割を引き受けます。

注 - 役割には、認証と特権コマンドが含まれます。役割の詳細については、『[Solaris のシステム管理: セキュリティーサービス](#)』の「RBAC の構成 (タスクマップ)」を参照してください。

- パッチをローカルディスクに保存する場合は、`/var/tmp/lupatches` のようなディレクトリを作成し、パッチをそのディレクトリにダウンロードします。
- [My Oracle Support](#) Web サイトから、パッチの一覧を取得します。
- パッチのディレクトリに移動します。

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

- `patchadd` コマンドを使用してパッチをインストールします。

```
# patchadd patch_id
```

`patch_id` はパッチの番号です。複数のパッチ名を指定する場合は、スペースで区切ります。

注 - パッチは、[My Oracle Support](#) のナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) に指定されている順序で適用する必要があります。

- 必要に応じてシステムをリブートします。いくつかのパッチは、有効にするためにリブートする必要があります。

x86 のみ: システムをリブートする必要があります。そうしないと、Live Upgrade は失敗します。

```
# init 6
```

2 ZFS ルートプールを作成します。

ZFS ルートプールは、ブート可能かつアップグレード可能にするため単一のスライス上に作成します。

```
# zpool create rpool c3t0d0s0
```

この例で作成する新しい ZFS の名前は `rpool` です。このプールが、ブート可能なスライス `c3t0d0s0` に作成されます。

新規ルートプールの作成の詳細については、『[Oracle Solaris ZFS 管理ガイド](#)』を参照してください。

3 UFS ルート (/) ファイルシステムを新しい ZFS ルートプールに移行します。

```
# lucreate [-c ufsBE] -n new-zfsBE -p rpool
```

- ufsBE** 現在の UFS ブート環境の名前。このオプションは必須ではなく、また、最初のブート環境の作成時にのみ使用します。lucreate コマンドを初めて実行する場合に -c オプションを省略すると、デフォルトの名前が自動的に作成されます。
- new-zfsBE** 作成されるブート環境の名前。名前は、システム上で一意となるように指定する必要があります。
- p rpool** 新しく作成した ZFS ルート (/) ファイルシステムを、rpool で定義した ZFS ルートプールに配置します。

すべての非共有非大域ゾーンは、クリティカルファイルシステムと共に、新しいブート環境にコピーされます。新しい ZFS ブート環境の作成には、しばらく時間がかかることがあります。UFS ファイルシステムのデータが ZFS ルートプールにコピーされます。非アクティブなブート環境が作成された場合は、luupgrade コマンドまたは luactivate コマンドを使用して、新しい ZFS ブート環境をアップグレードまたはアクティブにすることができます。

4 (オプション) ブート環境が完成したことを確認します。

lustatus コマンドは、ブート環境の作成が完了してブート可能であるかどうかを報告します。

```
# 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 (オプション) システム上の基本データセット情報を確認します。

list コマンドで、システム上のすべてのデータセットの名前を表示できます。この例では、rpool が ZFS プールの名前、new-zfsBE が新しく作成した ZFS ブート環境の名前です。

```
# zfs list
NAME                                USED  AVAIL  REFER  MOUNTPOINT
rpool                              11.4G  2.95G   31K    /rpool
rpool/ROOT                         4.34G  2.95G   31K    legacy
rpool/ROOT/new-zfsBE               4.34G  2.95G  4.34G    /
rpool/dump                         2.06G  5.02G   16K    -
rpool/swap                         5.04G  7.99G   16K    -
```

表示される新しいブート環境のマウントポイントは、luactivate コマンドが実行されるまでの一時的なものです。/dump ボリュームと /swap ボリュームは、元の UFS ブート環境と共有されませんが、ZFS ルートプール内およびルートプール内のブート環境内で共有されます。

例 13-1 非大域ゾーンがインストールされている UFS ルート (/) ファイルシステムから ZFS ルートプールへの移行

次の例では、既存の非大域ゾーン myzone の非大域ゾーンルートが、UFS ルート (/) ファイルシステムにあります。ゾーン zzone のゾーンルートは、既存の ZFS ストレージプール pool 内の ZFS ファイルシステムにあります。Live Upgrade を使用して、UFS ブート環境 c2t2d0s0 を ZFS ブート環境 zfs2BE に移行します。UFS ベースの myzone ゾーンが、Live Upgrade の操作前に作成された新しい ZFS ストレージプール mpool に移行されます。ZFS ベースの非大域ゾーン zzone はクローニングされるが、ZFS プール pool 内に保持され、新しい zfs2BE ブート環境に移行されます。

1. ブート環境を作成するコマンドは次のとおりです。

```
# 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 c4td0s0
# lucreate -c clt2d0s0 -n zfs2BE -p mpool
Checking GRUB menu...
Analyzing system configuration.
Updating boot environment description database on all BEs.
Updating system configuration files.
The device </dev/dsk/cltld0s0> is not a root device for any boot environment; cannot get BE ID.
Creating configuration for boot environment <zfs2BE>.
Source boot environment is <clt2d0s0>.
Creating file systems on boot environment <zfs2BE>.
Creating <zfs> file system for </> in zone <global> on <mpool/ROOT/zfs2BE>.
Populating file systems on boot environment <zfs2BE>.
Analyzing zones.
Mounting ABE <zfs2BE>.
Generating file list.
Copying data from PBE <clt2d0s0> to ABE <zfs2BE>.
100% of filenames transferred
Finalizing ABE.
Fixing zonpaths in ABE.
Unmounting ABE <zfs2BE>.
Fixing properties on ZFS datasets in ABE.
Reverting state of zones in PBE <clt2d0s0>.
Making boot environment <zfs2BE> bootable.
Updating bootenv.rc on ABE <zfs2BE>.
Saving existing file </boot/grub/menu.lst> in top level dataset for BE <zfs2BE> as <mount-point>
//boot/grub/menu.lst.prev.
File </boot/grub/menu.lst> propagation successful
Copied GRUB menu from PBE to ABE
No entry for BE <zfs2BE> in GRUB menu
Population of boot environment <zfs2BE> successful.
Creation of boot environment <zfs2BE> successful.
```

2. lucreate の操作が完了したら、この例で示すように lustatus コマンドを使ってブート環境のステータスを表示します。

```
# lustatus
Boot Environment      Is      Active Active      Can      Copy
Name                  Complete Now    On Reboot Delete 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
```

3. 次に、luactivate コマンドで新しい ZFS ブート環境をアクティブにします。例:

```
# luactivate zfsBE
A Live Upgrade Sync operation will be performed on startup of boot
environment <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. Boot the machine to Single User mode using a different boot device
(like the Solaris Install CD or Network). Examples:

    At the PROM monitor (ok prompt):
    For boot to Solaris CD: boot cdrom -s
    For boot to network:    boot net -s

3. Mount the Current boot environment root slice to some directory (like
/mnt). You can use the following command to mount:

    mount -Fufs /dev/dsk/clt0d0s0 /mnt

4. Run <luactivate> utility with out any arguments from the current boot
environment root slice, as shown below:

    /mnt/sbin/luactivate

5. luactivate, activates the previous working boot environment and
indicates the result.

6. Exit Single User mode and reboot the machine.
```

Modifying boot archive service
Activation of boot environment <zfsBE> successful.

4. システムをリブートして ZFS ブート環境にします。

```
# init 6
# svc.startd: The system is coming down. Please wait.
svc.startd: 79 system services are now being stopped.
.
.
.
```

5. この例のように、新しいブート環境および移行されたゾーンのステータスを確認します。

```
# 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	-

UFS ブート環境に戻す場合は、ZFS ブート環境で作成したすべての ZFS ストレージプールをもう一度インポートしてください。これらは、UFS ブート環境で自動的に使用可能にはならないからです。元の UFS ブート環境に切り替えると、次のようなメッセージが表示されます。

```
# luactivate clt2d0s0
WARNING: The following files have changed on both the current boot
environment <ZFSbe> zone <global> and the boot environment to be activated <clt2d0s0>:
/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 <clt2d0s0>. These files will not be automatically synchronized
from the current boot environment <ZFSbe> when boot environment <clt2d0s0>
```

パート III

付録

このパートでは、リファレンス情報について説明します。

Live Upgrade コマンドリファレンス

この付録では、コマンド行で入力できるコマンドの一覧を示します。Live Upgrade には、一覧で示すすべてのコマンド行ユーティリティのマニュアルページが含まれており、それらの一部についてはこのマニュアルでも説明しています。

タスク	コマンド
非アクティブブート環境をアクティブにします。	<code>luactivate(1m)</code>
スケジュールされた処理 (コピーまたは作成) を取り消します。	<code>lucancel(1M)</code>
アクティブブート環境を非アクティブブート環境と比較します。	<code>lucompare(1M)</code>
非アクティブブート環境を更新するためにファイルシステムをコピーし直します。	<code>lumake(1M)</code>
ブート環境を作成します。	<code>lucreate(1M)</code>
アクティブブート環境に名前を付けます。	<code>lucurr(1M)</code>
ブート環境を削除します。	<code>ludelete(1m)</code>
ブート環境の名前に記述を追加します。	<code>ludesc(1M)</code>
各ブート環境のクリティカルファイルシステムを表示します。	<code>lufslist(1m)</code>
ブート環境内のすべてのファイルシステムをマウントできるようにします。このコマンドを使用すると、ブート環境がアクティブでない時にそのブート環境内のファイルを変更できます。	<code>lumount(1M)</code>
ブート環境の名前を変更します。	<code>lurename(1M)</code>

タスク	コマンド
すべてのブート環境のステータスを表示します。	<code>lustatus(1M)</code>
ブート環境に存在するすべてのファイルシステムのマウントを解除します。このコマンドを使用すると、ブート環境がアクティブでない時にそのブート環境内のファイルを変更できます。	<code>luumount(1M)</code>
非アクティブブート環境上で OS をアップグレードするか、フラッシュアーカイブをインストールします。	<code>luupgrade(1M)</code>

トラブルシューティング(タスク)

この章では、Oracle Solaris 10 8/11 OS のインストール時に発生する可能性のあるエラーメッセージと一般的な問題の一覧を示します。それぞれの問題の解決方法を示します。内容は、インストールプロセスのどこで問題が発生したかに基づいてまとめられています。

- 227 ページの「ネットワークインストールの設定に関する問題」
- 228 ページの「システムのブートに関する問題」
- 234 ページの「Oracle Solaris OS の初期インストール」
- 236 ページの「Oracle Solaris OS のアップグレード」

注 - 「ブート可能なメディア」という語句は、Oracle Solaris インストールプログラムと、Oracle Solaris の機能である JumpStart のインストール方法を意味しています。

ネットワークインストールの設定に関する問題

Unknown client "*host-name*"

原因: `add_install_client` コマンドの *host-name* 引数は、ネームサービス内のホストではありません。

対処方法: ホスト *host_name* をネームサービスに追加し、`add_install_client` コマンドを実行し直してください。

Error: <system name> does not exist in the NIS ethers map

Add it, and rerun the `add_install_client` command

説明: `add_install_client` コマンドを実行すると、上記のエラーが出てコマンドが失敗します。

原因: インストールサーバーに追加しようとしているクライアントが、サーバーの `/etc/ethers` ファイルに存在しません。

対処方法: 必要な情報をインストールサーバー上の `/etc/ethers` ファイルに追加し、`add_install_client` コマンドを実行し直してください。

1. スーパーユーザーになるか、同等の役割を引き受けます。

注-役割には、認証と特権コマンドが含まれます。役割の詳細については、『*Solaris のシステム管理: セキュリティーサービス*』の「RBAC の構成 (タスクマップ)」を参照してください。

2. クライアント上で、`ethers` アドレスを見つけます。

```
# ifconfig -a grep ethers
ether 8:0:20:b3:39:1d
```

3. インストールサーバーで、`/etc/ethers` のリストにアドレスを追加します。
4. この例のように、クライアント上で `add_install_client` をもう一度実行します。

```
# ./add_install_client bluegill sun4u
```

システムのブートに関する問題

メディアからのブート時のエラーメッセージ

le0: No carrier - transceiver cable problem

原因: システムがネットワークに接続されていません。

対処方法: ネットワークに接続されていないシステムの場合は、このメッセージは無視してください。ネットワークに接続されているシステムの場合は、Ethernet ケーブルが正しく接続されていることを確認してください。

The file just loaded does not appear to be executable

原因: ブート用の適切な媒体が見つかりません。

対処方法: インストールサーバーからネットワークを介して Oracle Solaris 10 8/11 ソフトウェアをインストールするようにシステムが正しく設定されていることを確認します。

- Oracle Solaris Operating System DVD または Oracle Solaris Software CD のイメージをインストールサーバーにコピーした場合は、設定時にシステムのプラットフォームグループを正しく指定したかどうかを確認します。
- DVD または CD メディアを使用する場合は、Oracle Solaris Operating System DVD または Oracle Solaris Software - 1 CD がインストールサーバー上にマウントされていてアクセスできることを確認します。

boot: cannot open <filename> (SPARC ベースのシステムのみ)

原因: このエラーは、boot -file の場所を明示的に設定してオーバーライドするとき 발생합니다。

対処方法: 次のいずれかを試行します。

- PROM の boot -file を '' (無指定) に設定変更します。
- diag-switch が off と true に設定されていることを確認します。

Can't boot from file/device

原因: インストールメディアがブート可能なメディアを見つけることができません。

対処方法: 次の条件が満たされているか確認します。

- DVD-ROM または CD-ROM ドライブがシステムに適切に取り付けられ、電源が入っている。
- Oracle Solaris Operating System DVD または Oracle Solaris Software - 1 CD がドライブに挿入されている。
- ディスクに傷や埃が付いていない。

WARNING: clock gained xxx days -- CHECK AND RESET DATE! (SPARC システムのみ)

説明: これは参考情報です。

対処方法: メッセージは無視して、インストールを継続してください。

Not a UFS file system (x86 システムのみ)

原因: Oracle Solaris 10 8/11 ソフトウェアをインストールしたとき (Oracle Solaris インストールプログラムまたはカスタム JumpStart を使用)、ブートドライブを選択しませんでした。BIOS を編集してシステムをブートする必要があります。

対処方法: BIOS を選択してブートします。詳細は、BIOS のドキュメントを参照してください。

メディアからのブート時の一般的な問題

システムがブートしない。

説明: はじめて JumpStart サーバーを設定する場合、エラーメッセージを返さないブート問題が発生することがあります。システムに関する情報と、システムがブートする方法について確認するために、boot コマンドに -v オプションを付けて実行すると、詳細なデバッグ情報が表示されます。

注- このオプションを含めない場合、メッセージは出力されますが、システムログファイルが出力先となります。詳細は、[syslogd\(1M\)](#) のマニュアルページを参照してください。

対処方法: SPARC ベースシステムの場合、ok プロンプトで次のコマンドを入力します。

```
ok boot net -v - install
```

Toshiba SD-M 1401 DVD-ROM が搭載されたシステムで DVD メディアからのブートが失敗する

説明: 使用しているシステムにファームウェアバージョン 1007 の Toshiba SD-M1401 DVD-ROM が搭載されている場合、システムは Oracle Solaris Operating System DVD からブートできません。

対処方法: 111649-03 以降のパッチを適用して Toshiba SD-M1401 DVD-ROM ドライブのファームウェアを更新します。パッチ 111649-03 は、<http://support.oracle.com> (My Oracle Support) の「パッチと更新版」タブから入手できます。

メモリー増設用以外の PC カードを挿入すると、システムがハングまたはパニックを起こす。(x86 システムのみ)

原因: メモリー増設用以外の PC カードは、ほかのデバイスが使用するのと同じメモリーリソースを使用できません。

対処方法: この問題を解決するには、PC カードのドキュメントを参照してアドレス範囲を確認してください。

システムがプロンプトを出す前にハングする。(x86 システムのみ)

原因: サポートされていないハードウェアです。

対処方法: ハードウェアのドキュメントを参照してください。

ネットワークからのブート時のエラーメッセージ

WARNING: getfile: RPC failed: error 5 (RPC Timed out).

説明: インストールクライアントのブート要求に対して、ネットワーク上の複数のサーバーが応答したときに発生するエラーです。インストールクライアントの接続先のブートサーバーが間違っているため、インストールは停止します。次の原因が考えられます。

原因: 1 このインストールクライアントが登録された /etc/bootparams ファイルが複数のサーバーに存在する可能性があります。

対処方法:1 ネットワーク上の複数のサーバーの `/etc/bootparams` エントリにインストールクライアントが登録されていないか調べます。複数のサーバーに登録がされている場合は、インストールに使用するインストールサーバー（またはブートサーバー）以外のサーバーの `/etc/bootparams` ファイルから、クライアントの登録を削除します。

原因:2 複数の `/tftpboot` または `/rplboot` ディレクトリにこのインストールクライアントが登録されている可能性があります。

対処方法:2 ネットワーク上の複数のサーバーの `/tftpboot` または `/rplboot` ディレクトリにインストールクライアントが登録されていないか調べます。複数のサーバーに登録されている場合は、インストールに使用するインストールサーバー（またはブートサーバー）以外のサーバーの `/tftpboot` または `/rplboot` ディレクトリから、クライアントの登録を削除します。

原因:3 あるサーバーの `/etc/bootparams` ファイルにこのインストールクライアントが登録されており、別のサーバーの `/etc/bootparams` ファイルで、すべてのシステムがプロファイルサーバーにアクセスできるように記述されている可能性があります。エントリは次のようになります。

```
* install_config=profile-server:path
```

このエラーは、NIS または NIS+ の `bootparams` テーブルにこのような行が存在していても発生します。

対処方法:3 ワイルドカードエントリがネームサービスの `bootparams` マップまたはテーブル (`* install_config=` など) にある場合は、そのエントリを削除し、ブートサーバーの `/etc/bootparams` ファイルに追加します。

No network boot server. Unable to install the system. See installation instructions. (SPARC システムのみ)

原因:ネットワークからのインストールを試行しているシステムが、正しく設定されていません。

対処方法:ネットワークを介してインストールするようにシステムが適切に設定されているか確認します。『[Oracle Solaris 10 1/13 インストールガイド: ネットワークベースのインストール](#)』の「[CD イメージを使用してネットワークからインストールするシステムの追加](#)」を参照してください。

prom_panic: Could not mount file system (SPARC システムのみ)

原因:Oracle Solaris をネットワークからインストールしようとしています。が、ブートソフトウェアが次のものを見つけられません。

- Oracle Solaris Operating System DVD またはインストールサーバー上の Oracle Solaris Operating System DVD イメージコピー
- Oracle Solaris Software - 1 CD またはインストールサーバー上の Oracle Solaris Software - 1 CD イメージコピー

対処方法: インストール用のソフトウェアがマウントされ共有されるように設定してあることを確認します。

- インストールサーバーのDVD-ROM または CD-ROM ドライブから Oracle Solaris をインストールする場合は、Oracle Solaris Operating System DVD または Oracle Solaris Software - 1 CD が CD-ROM ドライブに挿入されて、マウントされており、`/etc/dfs/dfstab` ファイルで共有されていることを確認します。
- インストールサーバーのディスク上にある Oracle Solaris Operating System DVD イメージまたは Oracle Solaris Software - 1 CD イメージのコピーからインストールする場合は、そのコピーのディレクトリパスが `/etc/dfs/dfstab` ファイル内で共有されていることを確認します。

Timeout waiting for ARP/RARP packet... (SPARC システムのみ)

原因: 1 クライアントはネットワークからブートしようとしています、そのクライアントを認識するシステムを見つけることができません。

対処方法: 1 システムのホスト名が NIS または NIS+ のネームサービスに登録されていることを確認します。また、ブートサーバーの `/etc/nsswitch.conf` ファイル内の `bootparams` の検索順序を確認します。

たとえば、`/etc/nsswitch.conf` ファイル内にある次の行は、JumpStart または Oracle Solaris インストールプログラムが最初に NIS マップから `bootparams` 情報を探すことを示しています。ここで情報が見つからない場合、インストーラはブートサーバーの `/etc/bootparams` ファイルを調べます。

`bootparams: nis files`

原因: 2 クライアントの Ethernet アドレスが不正です。

対処方法: 2 インストールサーバーの `/etc/ethers` ファイルにあるクライアントの Ethernet アドレスが正しいことを確認します。

原因: 3 JumpStart インストールでは、`add_install_client` コマンドは特定のサーバーをインストールサーバーとして使用するプラットフォームグループを指定します。この問題が発生するのは、`add_install_client` を使用する際に不正なアーキテクチャー値を使用した場合です。たとえば、インストールするマシンが `sun4u` であるのに誤って `i86pc` を使用した場合などです。

対処方法: 3 正しいアーキテクチャー値を使用して `add_install_client` を実行し直します。

`ip: joining multicasts failed on tr0 - will use link layer broadcasts for multicast` (x86 システムのみ)

原因: このエラーメッセージは、トークンリングカードを使ってシステムをブートしたときに表示されます。Ethernet のマルチキャストとトークンリングのマルチキャストの動作は異なります。ドライバはこのエラーメッセージを返して、マルチキャストアドレスが無効なことを知らせます。

対処方法: このエラーメッセージは無視してください。マルチキャストがうまく動作しなければ、IP は代わりにレイヤーブロードキャストを使用し、インストールは失敗しません。

Requesting Internet address for *Ethernet-Address* (x86 ベースシステムのみ)

原因: クライアントはネットワークからブートしようとしていますが、そのクライアントを認識するシステムを見つけることができません。

対処方法: システムのホスト名がネームサービスに登録されていることを確認します。システムのホスト名がNISまたはNIS+のネームサービスに登録されているのに、システムがこのエラーメッセージを出力し続ける場合は、リブートしてみてください。

RPC: Timed out No bootparams (whoami) server responding; still trying... (x86 システムのみ)

原因: クライアントはネットワークからブートしようとしていますが、インストールサーバー上の `/etc/bootparams` ファイルにエントリを持つシステムを見つけることができません。

対処方法: インストールサーバー上で `add_install_client` を使用して、`/etc/bootparams` ファイルに適切なエントリを追加し、クライアントがネットワークからブートできるようにします。

Still trying to find a RPL server... (x86 システムのみ)

原因: システムはネットワークからブートしようとしていますが、サーバーではこのシステムをブートするように設定されていません。

対処方法: インストールサーバー上で、インストールするシステム用に `add_install_client` を実行します。`add_install_client` コマンドは、必要なネットワークブートプログラムを含む `/rplboot` ディレクトリを設定します。

CLIENT MAC ADDR: FF FF FF FF FF FF (DHCP によるネットワークインストールのみ)

原因: DHCP サーバーが正しく構成されていません。このエラーは、DHCP マネージャ内でオプションやマクロが正しく定義されていない場合に発生する可能性があります。

対処方法: DHCP マネージャで、オプションおよびマクロが正しく定義されていることを確認します。ルーターオプションが定義されており、その値がネットワークインストールで使用するサブネットを正しく表していることを確認します。

ネットワークからのブート時の一般的な問題

システムはネットワークを介してブートされるが、指定したインストールサーバー以外のシステムからブートされる。

原因: このクライアントが登録された `/etc/bootparams` エントリと `/etc/ethers` エントリが別のシステム上に存在します。

対処方法: ネームサーバー上で、インストールするシステムの `/etc/bootparams` エントリを更新します。このエントリは、次の構文に従う必要があります。

```
install-system root=boot-server:path install=install-server:path
```

また、サブネット内で複数のサーバーの `bootparams` ファイルにインストールクライアントが登録されていないか確認します。

システムがネットワークからブートしない (DHCP によるネットワークインストールのみ)。

原因: DHCP サーバーが正しく構成されていません。このエラーは、システムが DHCP サーバーのインストールクライアントとして構成されていない場合に発生することがあります。

対処方法: DHCP マネージャソフトウェアで、クライアントシステムのインストールオプションとマクロが定義されていることを確認します。詳細は、『[Oracle Solaris 10 1/13 インストールガイド: ネットワークベースのインストール](#)』の「[DHCP サービスによるシステム構成情報の事前構成 \(タスク\)](#)」を参照してください。

Oracle Solaris OS の初期インストール

Initial installation fails

対処方法: Oracle Solaris のインストールが失敗する場合、インストールを再実行する必要があります。インストールを再実行するには、Oracle Solaris Operating System DVD、Oracle Solaris Software - 1 CD、またはネットワークを利用してシステムをブートする必要があります。

Oracle Solaris ソフトウェアが部分的にインストールされたあとでは、このソフトウェアをアンインストールすることはできません。バックアップからシステムを復元するか、Oracle Solaris インストールの処理をもう一度開始する必要があります。

`/cdrom/cdrom0/SUNW xxxx/reloc.cpio: Broken pipe`

説明: このエラーメッセージは参考情報であり、インストールには影響しません。パイプへ書き込みをしたときに読み取りプロセスが存在しないと、この状況が発生します。

対処方法: メッセージは無視して、インストールを継続してください。

WARNING: CHANGE DEFAULT BOOT DEVICE (x86 システムのみ)

原因: これは参考情報です。システムの BIOS に設定されているデフォルトブートデバイスが、ブート時に Oracle Solaris Device Configuration Assistant を必要とするように設定されている可能性があります。

対処方法: インストールを続行します。Oracle Solaris Device Configuration Assistant を必要としないデバイスに Oracle Solaris ソフトウェアをインストールし終わったら、必要に応じて、BIOS に指定されたシステムのデフォルトのブートデバイスを変更します。

x86 のみ -locale キーワードを使用して初期インストール用の JumpStart プロファイルをテストする場合は、`pfinstall -D` コマンドでプロファイルをテストすると失敗します。回避方法については、[236 ページの「Oracle Solaris OS のアップグレード」](#) セクションのエラーメッセージ「could not select locale」を参照してください。

▼ x86: IDE ディスクの不良ブロックを検査する方法

IDE ディスクドライブは、Oracle Solaris ソフトウェアがサポートするほかのドライブのように、不良ブロックを自動的にマップアウトしません。IDE ディスク上に Oracle Solaris をインストールする前に、ディスクの表面解析を実行することをお勧めします。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
役割には、認証と特権コマンドが含まれます。役割の詳細については、『Solaris のシステム管理: セキュリティサービス』の「RBAC の構成 (タスクマップ)」を参照してください。
- 2 インストールメディアに応じた方法でブートします。
- 3 インストールタイプの選択を求めるプロンプトが表示されたら、オプション 6「Single user shell」を選択します。
- 4 **format(1M)** プログラムを起動します。
`format`
- 5 ディスク面の検査をする IDE ディスクドライブを指定します。
`cxdy`
cx コントローラ番号
dy デバイス番号

- 6 **fdisk** パーティションがあるかどうかを判定します。

Oracle Solaris **fdisk** パーティションが存在しない場合は、**fdisk** コマンドを使用してディスク上に作成します。

```
format> fdisk
```

- 7 表面解析を開始します。

```
format> analyze
```

- 8 現在の設定を判定します。

```
analyze> config
```

- 9 (省略可能) 設定を変更します。

```
analyze> setup
```

- 10 不良ブロックがあるかどうか判定します。

```
analyze> type-of-surface-analysis
```

type-of-surface-analysis read、write、または compare

format が不良ブロックを発見すると、それらの再マッピングを実行します。

- 11 解析を終了します。

```
analyze> quit
```

- 12 必要な場合、再マッピングするブロックを指定します。

```
format> repair
```

- 13 **format** プログラムを終了します。

```
quit
```

- 14 マルチユーザーモードでメディアを再起動します。

```
# exit
```

Oracle Solaris OS のアップグレード

アップグレード時のエラーメッセージ

No upgradable disks

原因: /etc/vfstab ファイルのスワップエントリが原因でアップグレードに失敗しました。

対処方法: /etc/vfstab ファイルの次の行をコメントにします。

- アップグレードしないディスク上のスワップファイルとスライスを指定している行
- 存在しないスワップファイルを指定している行
- 使用していないスワップスライスを指定している行

usr/bin/bzcat not found

原因: パッチクラスタが必要なために Live Upgrade が失敗しています。

対処方法: Live Upgrade をインストールするにはパッチが必要です。 <http://support.oracle.com> (My Oracle Support) を調べて最新の更新済みパッチリストであることを確認してください。My Oracle Support で、ナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) を検索してください。

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 システムのみ)

原因: 十分な容量がないため、Oracle Solaris Software - 1 CD でアップグレードできません。

対処方法: アップグレードするには、512M バイト以上のスワップスライスを作成するか、または Oracle Solaris Operating System DVD の Oracle Solaris のインストール、ネットインストールイメージ、JumpStart などの別のアップグレード方法を選択します。

ERROR: Could not select locale (x86 システムのみ)

原因: pfinstall -D コマンドを使用して JumpStart プロファイルをドライランテストするとき、次のような条件下ではテストが失敗します。

- プロファイルに locale キーワードが含まれている。
- GRUB ソフトウェアが含まれているリリースをテストしている。Solaris 10 1/06 以降のリリースでは、GRUB ブートローダーにより、システムにインストールされているさまざまなオペレーティングシステムを GRUB メニューで簡単にブートできます。

GRUB ソフトウェアの導入に伴い、ミニルートは圧縮されています。ソフトウェアでは、圧縮されたミニルートからロケールのリストを見つけることができなくなりました。ミニルートはもっとも小さい Oracle Solaris ルート (/) ファイルシステムで、Oracle Solaris インストールメディアにあります。

対処方法: 次の手順を実行します。次の値を使用してください。

- MEDIA_DIR は /cdrom/cdrom0/

- MINIRoot_DIR は \$MEDIA_DIR /Solaris_10/Tools/Boot
 - MINIRoot_ARCHIVE は \$MEDIA_DIR/boot/x86.miniroot
 - TEMP_FILE_NAME は /tmp/test
1. スーパーユーザーになるか、同等の役割を引き受けます。
 役割には、認証と特権コマンドが含まれます。役割の詳細については、『Solaris のシステム管理: セキュリティーサービス』の「RBAC の構成 (タスクマップ)」を参照してください。
 2. ミニルートアーカイブを圧縮解除します。

```
# /usr/bin/gzcat $MINIRoot_ARCHIVE > $TEMP_FILE_NAME
```
 3. lofiadm コマンドを使用して、ミニルートデバイスを作成します。

```
# LOFI_DEVICE=/usr/sbin/lofiadm -a $TEMP_FILE_NAME
# echo $LOFI_DEVICE
/dev/lofi/1
```
 4. lofi コマンドを使用して、ミニルートディレクトリの下にミニルートをマウントします。

```
# /usr/sbin/mount -F ufs $LOFI_DEVICE $MINIRoot_DIR
```
 5. プロファイルをテストします。

```
# /usr/sbin/install.d/pfinstall -D -c $MEDIA_DIR $path-to-jumpstart_profile
```
 6. テストが完了したら、lofi デバイスのマウントを解除します。

```
# umount $LOFI_DEVICE
```
 7. lofi デバイスを削除します。

```
# lofiadm -d $TEMP_FILE_NAME
```

アップグレード時の一般的な問題

システム上にアップグレード可能なバージョンの Solaris ソフトウェアが存在するにもかかわらず、アップグレードオプションが提供されない。

原因: 1 /var/sadm ディレクトリがシンボリックリンクであるか、別のファイルシステムからマウントされたディレクトリです。

対処方法: 1 /var/sadm ディレクトリをルート (/) または /var ファイルシステムに移動します。

原因: 2 /var/sadm/softinfo/INST_RELEASE ファイルが存在しません。

対処方法: 2 次の形式で新しく INST_RELEASE ファイルを作成します。

OS=Solaris
 VERSION=x
 REV=0

x システム上の Oracle Solaris ソフトウェアのバージョン

原因: /var/sadm/softinfo に SUNWusr パッケージが存在しません。

対処方法: 3 初期インストールを行う必要があります。Oracle Solaris ソフトウェアはアップグレードできません。

md ドライバの停止または初期化に失敗する

対処方法: 次の手順を実行します。

- ファイルシステムが RAID-1 ボリュームでない場合は、vsftab ファイル内でコメントにします。
- ファイルシステムが RAID-1 ボリュームであれば、ミラーを解除し、インストールし直します。ミラー化の解除については、『[Solaris Volume Manager 管理ガイド](#)』の「[RAID-1 ボリュームの削除\(ミラー化の解除\)](#)」を参照してください。

Solaris インストールプログラムがファイルシステムをマウントできないため、アップグレードに失敗する。

原因: アップグレード時に、スクリプトは、アップグレード対象のルート (/) ファイルシステム上に、システムの /etc/vfstab ファイルに記載されているすべてのファイルシステムをマウントしようとします。インストールプログラムがファイルシステムをマウントできない場合、失敗して終了します。

対処方法: システムの /etc/vfstab ファイル内のすべてのファイルシステムがマウントできることを確認します。/etc/vfstab ファイル内のマウントできない、または問題の原因になっている可能性があるファイルシステムをすべてコメントにすると、Oracle Solaris インストールプログラムはアップグレード中にコメントにしたファイルシステムをマウントしません。アップグレードされるソフトウェアを含む、システムベースのファイルシステム(たとえば /usr)はコメントにできません。

The upgrade fails

説明: システムにアップグレードに対応できるだけの十分なディスク容量がありません。

原因: 『[Oracle Solaris 10 1/13 インストールガイド: インストールとアップグレードの計画](#)』の「[ディスク容量の再配置を伴うアップグレード](#)」で領域の要件を確認し、自動配置機能を使用して領域の再配置を行わずにこの問題を解決できるかどうかを調べます。

RAID-1 ボリュームのルート (/) ファイルシステムのアップグレードに関連する問題
対処方法: ルート (/) ファイルシステムが Solaris ボリュームマネージャーの RAID-1
ボリュームである場合に、アップグレードの問題が発生するときは、『[Solaris
Volume Manager 管理ガイド](#)』の第 25 章「[Solaris Volume Manager のトラブル
シューティング \(タスク\)](#)」を参照してください。

▼ アップグレード失敗後にアップグレードを継続する方法

電源障害やネットワーク接続の失敗など、制御できない理由でアップグレードが失敗し、システムをソフトブートできない場合は、アップグレードの続行を試行してください。

- 1 **Oracle Solaris Operating System DVD、Oracle Solaris Software - 1 CD、またはネットワーク**を利用してシステムをリブートします。
- 2 インストール用のアップグレードオプションを選択します。
Oracle Solaris インストールプログラムは、システムが部分的にアップグレードされているかどうかを判定し、アップグレードを継続します。

x86: GRUB を使用する場合の Live Upgrade の問題

x86 ベースシステムで Live Upgrade と GRUB ブートローダーを使用すると、次のようなエラーが発生する可能性があります。

ERROR: The media product tools installation directory *path-to-installation-directory* does not exist.

ERROR: The media *dirctory* does not contain an operating system upgrade image.

説明: これらのエラーメッセージは、新しいブート環境をアップグレードするために `luupgrade` コマンドを使用するときに発生することがあります。

原因: 古いバージョンの Live Upgrade が使用されています。システムにインストールした Live Upgrade パッケージは、メディアおよびメディア上のリリースと互換性がありません。

対処方法: Live Upgrade パッケージは、常にアップグレード後のリリースのものを使用してください。

例: 次の例のエラーメッセージは、システムの Live Upgrade パッケージのバージョンが、メディアのパッケージのバージョンと異なることを示しています。


```
# 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.
```

ERROR: Cannot find or is not executable: </sbin>.

ERROR: One or more patches required by Live Upgrade has not been installed.

原因: Live Upgrade で必要とされる 1 つ以上のパッチが、システムにインストールされていません。このエラーメッセージでは、欠落しているすべてのパッチを認識しているわけではありません。

対処方法: Live Upgrade を使用する前に、必要なパッチすべてを必ずインストールしてください。 <http://support.oracle.com> (My Oracle Support) を調べて最新の更新済みパッチリストであることを確認してください。My Oracle Support で、ナレッジドキュメント 1004881.1 - Live Upgrade Software Patch Requirements (以前の 206844) を検索してください。

ERROR: Device mapping command </sbin/biosdev> failed. Please reboot and try again.

原因: 1 Live Upgrade が、以前の管理タスクが原因でデバイスをマップできません。

対処方法: 1 システムをリブートして、もう一度 Live Upgrade を実行します

原因: 2 システムをリブートしても同じエラーメッセージが表示される場合は、2 つ以上の同一ディスクがあります。デバイスのマッピングコマンドがそれらのディスクを区別できません。

対処方法: 2: ディスクの 1 つにダミーの新しい fdisk パーティションを作成し、システムをリブートします。詳細は、 [fdisk\(1M\)](#) のマニュアルページを参照してください。

Cannot delete the boot environment that contains the GRUB menu

原因: Live Upgrade には、ブート環境に GRUB メニューが含まれる場合はブート環境を削除できないという制限があります。

対処方法: [lumake\(1M\)](#) コマンドまたは [luupgrade\(1M\)](#) コマンドを使用してブート環境を再使用します。

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.

原因: GRUB メニューを含むファイルシステムは、システムをブート可能な状態に維持するために不可欠です。Live Upgrade コマンドは、GRUB メニューを破棄しま

せん。ただし、Live Upgrade コマンド以外のコマンドを使用して GRUB メニューのあるファイルシステムを誤って再作成または破棄すると、回復ソフトウェアは GRUB メニューの再インストールを試みます。回復ソフトウェアは、次のリブート時に GRUB メニューを同じファイルシステムに戻します。たとえば、ファイルシステムで `newfs` または `mkfs` コマンドを使用し、誤って GRUB メニューを破棄してしまったとします。GRUB メニューを正しく復元するには、スライスが次の条件を満たす必要があります。

- マウント可能なファイルシステムが含まれている
- スライスが以前に存在していた Live Upgrade ブート環境の一部である

システムをリブートする前に、必要であればスライスを修正します。

対処方法: システムをリブートします。GRUB メニューのバックアップコピーが自動的にインストールされます。

The GRUB menu's menu.lst file was accidentally deleted.

対処方法: システムをリブートします。GRUB メニューのバックアップコピーが自動的にインストールされます。

Veritas VxVm の実行中に Live Upgrade を使用してアップグレードするとシステムパニックが発生する

▼ VERITAS VxVm の実行中にアップグレードする方法

Veritas VxVM の実行中に Live Upgrade を使用してアップグレードを行う場合、次の手順でアップグレードを行わないと、リブート時にシステムパニックが発生します。この問題は、パッケージが Oracle Solaris の最新のパッケージガイドラインに従っていない場合に発生します。

- 1 スーパーユーザーになるか、同等の役割を引き受けます。
役割には、認証と特権コマンドが含まれます。役割の詳細については、『Solaris のシステム管理: セキュリティーサービス』の「RBAC の構成 (タスクマップ)」を参照してください。
- 2 非アクティブブート環境を作成します。66 ページの「新しいブート環境の作成」を参照してください。

- 3 非アクティブブート環境をアップグレードする前に、非アクティブブート環境上の既存の Veritas ソフトウェアを無効にします。

- a. 非アクティブブート環境をマウントします。

```
# lumount inactive-boot-environment-name mount-point
```

例:

```
# lumount solaris8 /mnt
```

- b. **vfstab** ファイルが含まれているディレクトリに変更します。

例:

```
# cd /mnt/etc
```

- c. 非アクティブブート環境の **vfstab** ファイルをコピーします。

例:

```
# cp vfstab vfstab.501
```

- d. コピーされた **vfstab** 内のすべての Veritas ファイルシステムエントリをコメントにします。

例:

```
# sed '/vx\/dsk\/s\/^\/#/g' < vfstab > vfstab.novxfs
```

各行の最初の文字が # に変わり、その行がコメント行になります。このコメント行は、system ファイルのコメント行とは異なります。

- e. 変更した **vfstab** ファイルをコピーします。

例:

```
# cp vfstab.novxfs vfstab
```

- f. 非アクティブブート環境の **system** ファイルがあるディレクトリに変更します。

例:

```
# cd /mnt/etc
```

- g. 非アクティブブート環境の **system** ファイルをコピーします。

例:

```
# cp system system.501
```

- h. **drv/vx** を含むすべての **forceload:** エントリをコメントにします。

```
# sed '/forceload: drv\/vx\/s\/^\/*/' <system> system.novxfs
```

各行の最初の文字が * に変わり、その行がコメント行になります。このコメント行は、vfstab ファイルのコメント行とは異なります。

- i. **Veritas install-db** ファイルを作成します。
例:

```
# touch vx/reconfig.d/state.d/install-db
```
- j. 非アクティブブート環境のマウントを解除します。

```
# luumount inactive-boot-environment-name
```
- 4 非アクティブブート環境をアップグレードします。
第5章「[Live Upgrade によるアップグレード \(タスク\)](#)」を参照してください。
- 5 非アクティブブート環境をアクティブにします。
[113 ページの「ブート環境のアクティブ化」](#)を参照してください。
- 6 システムをシャットダウンします。

```
# init 0
```
- 7 非アクティブブート環境をシングルユーザーモードでブートします。

```
OK boot -s
```

vxvm または VXVM を含むメッセージとエラーメッセージがいくつか表示されますが、これらは無視してかまいません。非アクティブブート環境がアクティブになります。
- 8 **Veritas** をアップグレードします。
 - a. 次の例のように、システムから **Veritas VRTSvmsa** パッケージを削除します。

```
# pkgrm VRTSvmsa
```
 - b. **Veritas** パッケージがあるディレクトリに移動します。

```
# cd /location-of-Veritas-software
```
 - c. システムに最新の **Veritas** パッケージを追加します。

```
# pkgadd -d 'pwd' VRTSvxvm VRTSvmsa VRTSvmdoc VRTSvmman VRTSvmdev
```
- 9 元の **vfstab** と **system** ファイルを復元します。

```
# cp /etc/vfstab.original /etc/vfstab  
# cp /etc/system.original /etc/system
```
- 10 システムをリブートします。

```
# init 6
```

x86: 既存のサービスパーティションが存在しないシステムでは、デフォルトでサービスパーティションが作成されない

診断・サービスパーティションの存在しないシステム上に Oracle Solaris 10 8/11 OS をインストールすると、インストールプログラムがデフォルトでサービスパーティションを作成しない場合があります。Oracle Solaris パーティションと同じディスクにサービスパーティションを含める場合は、Oracle Solaris 10 8/11 OS をインストールする前にサービスパーティションを作り直す必要があります。

サービスパーティションが存在しているシステムに Solaris 8 2/02 OS をインストールした場合、インストールプログラムがサービスパーティションを保持しなかった可能性があります。サービスパーティションを保持するように fdisk ブートパーティションレイアウトを手動で編集しなかった場合、インストールプログラムはインストール時にサービスパーティションを削除しています。

注 - Solaris 8 2/02 OS のインストール時にサービスパーティションを明確に保持しなかった場合、サービスパーティションを作り直して Oracle Solaris 10 8/11 OS にアップグレードすることができない可能性があります。

Oracle Solaris パーティションが含まれているディスクにサービスパーティションを含める場合は、次のいずれかの方法を選択してください。

▼ ソフトウェアをネットワークインストールイメージまたは **Oracle Solaris Operating System DVD** からインストールするときにサービスパーティションを含める方法

ソフトウェアを、ネットインストールイメージからインストールするか、ネットワーク経由で Oracle Solaris Operating System DVD からインストールする場合、次の手順を実行します。

- 1 ディスクの内容を削除します。
- 2 インストールする前に、システムの診断用 CD を使用してサービスパーティションを作成します。
サービスパーティションの作成方法の詳細は、ハードウェアのドキュメントを参照してください。

- 3 ネットワークからシステムをブートします。
「fdisk パーティションのカスタマイズ」画面が表示されます。
- 4 「デフォルト」をクリックし、デフォルトのブートディスクパーティションレイアウトを読み込みます。
インストールプログラムにより、サービスパーティションが保持され、Oracle Solaris パーティションが作成されます。

▼ ソフトウェアを **Oracle Solaris Software - 1 CD** またはネットワークインストールイメージからインストールするときにサービスパーティションを含める方法

Oracle Solaris インストールプログラムを使用して、Oracle Solaris Software - 1 CD またはブートサーバー上のネットワークインストールイメージからインストールを実行するには、次の手順を実行します。

- 1 ディスクの内容を削除します。
- 2 インストールする前に、システムの診断用 **CD** を使用してサービスパーティションを作成します。
インストールプログラムにより、Oracle Solaris パーティションの作成方法を選択するよう求められます。サービスパーティションの作成方法の詳細は、ハードウェアのドキュメントを参照してください。
- 3 システムをブートします。
- 4 「残りのディスクを使用して **Solaris** パーティションを配置します」を選択します。
インストールプログラムにより、サービスパーティションが保持され、Oracle Solaris パーティションが作成されます。
- 5 インストールを完了します。

その他の SVR4 パッケージ要件 (リファレンス)

この付録は、パッケージ (特にサードパーティーパッケージ) をインストールまたは削除するシステム管理者を対象としています。パッケージに関するこれらの要件に従うことで、次の利点が得られます。

- 現在稼働中のシステムは変更されず、Live Upgrade によるアップグレードと、非大域ゾーンおよびディスクレスクライアントの作成と管理が可能です。
- JumpStart などのインストールプログラムを使用する場合に、パッケージを対話式ではなく自動でインストールできます。

この章には次のセクションが含まれています。

- 247 ページの「稼働中の OS に対する変更の防止」
- 251 ページの「インストール中およびアップグレード中のユーザー操作の回避」
- 252 ページの「ゾーンのパッケージパラメータの設定」

稼働中の OS に対する変更の防止

このセクションで説明する要件に従えば、現在稼働中の OS は変更されません。

絶対パスの使用

オペレーティングシステムのインストールに成功するためには、Live Upgrade の非アクティブブート環境などの代替ルート (/) ファイルシステムをパッケージが認識して、正しく従う必要があります。

パッケージの `pkgmap` ファイル (パッケージマップ) には、絶対パスを指定できます。これらのパスが存在する場合、そのファイルは、`pkgadd` コマンドの `-R` オプションとの相対パスに書き込まれます。絶対パスと相対 (再配置可能) パスの両方を含むパッケージは、代替ルート (/) ファイルシステムにもインストールできます。絶

対ファイルであれ再配置可能ファイルであれ、その前には `$PKG_INSTALL_ROOT` が付加されるため、`pkgadd` によるインストールでは、すべてのパスが正しく解釈されます。

pkgadd -R コマンドの使用

`pkgadd -R` オプションを使ってパッケージをインストールしたり、`pkgrm -R` オプションを使ってパッケージを削除する場合には、パッケージが、動作中のシステムを変更してはいけません。この機能は、JumpStart、Live Upgrade、非大域ゾーン、およびディスクレスクライアントで使用されます。

さらに、`pkgadd -R` オプションを使ってパッケージをインストールしたり、`pkgrm -R` オプションを使ってパッケージを削除する場合には、パッケージに同梱のスクリプトで、動作中のシステムを変更してはいけません。インストールスクリプトを作成する場合には、参照するディレクトリやファイルの前に `$PKG_INSTALL_ROOT` 変数を付加する必要があります。パッケージでは、書き込むすべてのディレクトリやファイルの前に `$PKG_INSTALL_ROOT` を付加する必要があります。さらに、パッケージでは、`$PKG_INSTALL_ROOT` 接頭辞を付加せずにディレクトリを削除すべきではありません。

次の表に、スクリプト構文の例を示します。

表 C-1 インストールスクリプト構文の例

スクリプトタイプ	正しい構文	正しくない構文
Bourne シェル「if」ステートメントの一部	<pre>if [-f \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf] ; then</pre>	<pre>if [-f /etc/myproduct.conf] ; \ then</pre>
ファイルの削除	<pre>/bin/rm -f \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf</pre>	<pre>/bin/rm -f /etc/myproduct.conf</pre>
ファイルの変更	<pre>echo "test=no" > \${PKG_INSTALL_ROOT}\ /etc/myproduct.conf</pre>	<pre>echo "test=no" > \ /etc/myproduct.conf</pre>

\$PKG_INSTALL_ROOT と \$BASEDIR の相違点

`$PKG_INSTALL_ROOT` は、パッケージを追加しようとするマシンのルート (/) ファイルシステムの場所です。この値は、`pkgadd` コマンドの `-R` 引数の後にセットされます。たとえば、次のコマンドを実行すると、パッケージのインストール時に `$PKG_INSTALL_ROOT` の値は `/a` になります。

```
# pkgadd -R /a SUNWvxvm
```


\$BASEDIR は、再配置可能なパッケージオブジェクトがインストールされる「再配置可能」なベースディレクトリを指しています。ここにインストールされるのは、再配置可能オブジェクトだけです。再配置可能でないオブジェクト (pkgmap ファイルに「絶対」パスが指定されているオブジェクト) は、必ず、非アクティブブート環境からの相対位置にインストールされます。\$BASEDIR からの相対位置ではありません。再配置可能なオブジェクトがないパッケージは、絶対パッケージ (再配置不可) と呼ばれます。その場合、\$BASEDIR は未定義であるため、これをパッケージに添付されているスクリプトで使用することはできません。

たとえば、パッケージの pkgmap ファイルに次のエントリがあるとします。

```
1 f none sbin/ls 0555 root sys 3541 12322 1002918510
1 f none /sbin/ls2 0555 root sys 3541 12322 2342423332
```

さらに、pkginfo ファイルには、\$BASEDIR が次のように指定されているとします。

```
BASEDIR=/opt
```

このパッケージを次のコマンドでインストールすると、ls は、/a/opt/sbin/ls としてインストールされますが、ls2 は、/a/sbin/ls2 としてインストールされます。

```
# pkgadd -R /a SUNWtest
```

スクリプト作成のガイドライン

パッケージ処理のスクリプトを作成するときは、現在稼働中の OS への変更を防ぐために、OS に依存しないようにしてください。スクリプトには、パッケージのインストールや削除の実行中に行うアクションを定義します。事前に決められたプロシージャ名で作成できるスクリプトが4つあります。preinstall、postinstall、preremove、および postremove です。

表 C-2 スクリプト作成のガイドライン

ガイドライン	Live Upgrade への影響	非大域ゾーン への影響
スクリプトは Bourne シェル (/bin/sh) で書き込む必要があります。pkgadd コマンドは、スクリプトの実行時にインタプリタとして Bourne シェルを使用します。	X	X
スクリプトはプロセスの開始や停止を行ったり、ps や truss などのコマンドの出力に依存したりしてはいけません。ps や truss はオペレーティングシステムに依存し、稼働中のシステムに関する情報を報告します。	X	X
スクリプトでは、expr、cp、ls などの標準的な UNIX コマンドや、シェルスクリプトの作成を容易にするその他のコマンドを自由に使用できます。	X	X

表 C-2 スクリプト作成のガイドライン (続き)

ガイドライン	Live Upgrade への影響	非大域ゾーン への影響
パッケージはサポートされているすべてのリリースで動作する必要があるため、スクリプト X で呼び出すコマンドはこれらすべてのリリースで利用可能である必要があります。したがって、Solaris 8 リリースのあとで追加または削除されたコマンドは使用できません。		
特定のコマンドまたはオプションが Solaris 8、Solaris 9、または Oracle Solaris 10 リリースでサポートされていることを確認するには、 http://www.oracle.com/technetwork/indexes/documentation/index.html にある個別のリファレンスマニュアルバージョンを参照してください。		

ディスクレスクライアントの互換性維持

パッケージでは、パッケージ自体が提供しているコマンドを実行してはいけません。この制限により、ディスクレスクライアントの互換性を維持し、まだインストールされていない共有ライブラリを必要とする可能性があるコマンドの実行を防ぎます。

パッケージの検証

すべてのパッケージは `pkgchk` の検証にパスしなければなりません。パッケージを作成したらインストールする前に、次のコマンドでパッケージをチェックする必要があります。

```
# pkgchk -d dir-name pkg-name
dir-name   パッケージがあるディレクトリの名前を指定します。
pkg-name   パッケージの名前を指定します。
```

例 C-1 パッケージをテストする

パッケージを作成したら、`pkgadd` で `-R dir-name` オプションを使用して、これを代替ルート (*/*) ファイルシステムにインストールしてテストする必要があります。パッケージをインストールしたら、次のように、パッケージが正しいかどうかを `pkgchk` コマンドでチェックします。

```
# pkgadd -d . -R /a SUNWvxxm
# pkgchk -R /a SUNWvxxm

エラーが表示されないことを確認します。
```

例 C-2 /export/SUNWvxvmにあるパッケージをテストする

たとえば、パッケージが /export/SUNWvxvmにある場合は、次のコマンドを実行します。

```
# pkgchk -d /export SUNWvxvm
```

エラーが表示されないことを確認します。

ファイルの作成、変更、削除を行うときに、ほかのコマンドでパッケージを検証することもできます。例:

- `dircmp` または `fssnap` コマンドを使用して、パッケージが正しく動作するかどうかを検証できます。
- `ps` コマンドを使用して、パッケージによりデーモンの開始や停止が行われていないことを確認することで、デーモンの要件準拠をテストできます。
- `truss`、`pkgadd -v`、および `pkgrm` コマンドで、実行時のパッケージインストールの要件準拠をテストできますが、これが常に機能するとはかぎりません。次の例では、`truss` コマンドは、読み取り専用ディレクトリおよび `$TMPDIR` 以外のディレクトリへのアクセス情報をすべて除外し、指定された非アクティブブート環境以外のディレクトリへの読み取り専用でないアクセス情報のみを表示します。

```
# TMPDIR=/a; export TMPDIR
# truss -t open /usr/sbin/pkgadd -R ${TMPDIR} SUNWvxvm \
2>&1 > /dev/null | grep -v O_RDONLY | grep -v \
'open("${TMPDIR}
```

インストール中およびアップグレード中のユーザー操作の回避

次の Oracle Solaris 標準ユーティリティを使用するときは、ユーザーが情報の入力を求められることなく、パッケージの追加や削除が行われる必要があります。

- JumpStart プログラム
- Live Upgrade
- Oracle Solaris のインストール プログラム
- Oracle Solaris ゾーン

パッケージをテストして、ユーザー操作なしでインストールされるようにするには、`pkgadd -a` コマンドで新しい管理ファイルを設定します。`-a` オプションは、デフォルトの管理ファイルの代わりにユーザー定義の管理ファイルを使用することを意味します。デフォルトのファイルを使用すると、情報の入力が必要になることがあります。管理ファイルを作成すれば、`pkgadd` でこのようなチェックを省略

し、ユーザーの確認なしでパッケージをインストールすることができます。詳細は、[admin\(4\)](#) または [pkgadd\(1M\)](#) のマニュアルページを参照してください。

次の例では、`pkgadd` コマンドによる管理ファイルの扱いを示します。

- 管理ファイルを指定しないと、`pkgadd` は `/var/sadm/install/admin/default` を使用します。このファイルを使用すると、ユーザーの入力が必要になることがあります。

```
# pkgadd
```

- コマンド行に相対的な管理ファイルを指定すると、`pkgadd` は `/var/sadm/install/admin` からこのファイル名を探して使用します。この例では相対的な管理ファイルの名前が `nocheck` であるため、`pkgadd` は `/var/sadm/install/admin/nocheck` を探します。

```
# pkgadd -a nocheck
```

- 絶対パスでファイルを指定すると、`pkgadd` はこれを使用します。この例では、`pkgadd` は `/tmp` 内で `nocheck` 管理ファイルを検索します。

```
# pkgadd -a /tmp/nocheck
```

例C-3 インストール管理ファイル

次の例では、`pkgadd` ユーティリティーでユーザーの入力をほとんど必要としないインストール管理ファイルを示します。パッケージがシステムで利用可能な容量を超えた容量を必要としない限り、`pkgadd` ユーティリティーはこのファイルを使用して、ユーザーに情報の入力を求めることなくインストールを実行します。

```
mail=
instance=overwrite
partial=nocheck
runlevel=nocheck
idepend=nocheck
space=ask
setuid=nocheck
conflict=nocheck
action=nocheck
basedir=default
```

ゾーンのパッケージパラメータの設定

パッケージが保持するパラメータにより、非大域ゾーンのインストールされたシステムで内容を配布および可視にする方法が制御されます。

`SUNW_PKG_ALLZONES`、`SUNW_PKG_HOLLOW`、および `SUNW_PKG_THISZONE` パッケージパラメータは、ゾーンがインストールされているシステムでのパッケージの特性を定義します。非大域ゾーンのインストールされたシステムでパッケージを管理できるようにするには、これらのパラメータを設定する必要があります。

表 C-3 に、パッケージパラメータ設定のための 4 つの有効な組み合わせを示します。この表に示されていない設定の組み合わせを選択すると、そのような設定は無効であり、パッケージのインストールは失敗します。

注-3 つのパッケージパラメータをすべて設定したことを確認してください。3 つのパッケージパラメータをすべて空のままにしてもかまいません。ゾーンのパッケージパラメータが見つからない場合、パッケージツールでは `false` の設定として解釈されますが、パラメータの設定を省略しないように強くお勧めします。3 つのパッケージパラメータをすべて設定することにより、パッケージをインストールまたは削除するときのパッケージツールの動作を正確に指定します。

表 C-3 ゾーンのパッケージパラメータの有効な設定

SUNW_PKG_ALLZONES の設定	SUNW_PKG_HOLLOW の設定	SUNW_PKG_THISZONE の設定	パッケージの説明
false	false	false	<p>ゾーンのパッケージパラメータのすべてに値を指定しない、パッケージに対するデフォルト設定です。</p> <p>この設定を持つパッケージは、大域ゾーンまたは非大域ゾーンにインストールできます。</p> <ul style="list-style-type: none"> ■ 大域ゾーン内で <code>pkgadd</code> コマンドを実行すると、パッケージは大域ゾーンおよびすべての非大域ゾーンにインストールされます。 ■ 非大域ゾーン内で <code>pkgadd</code> コマンドを実行すると、パッケージはその非大域ゾーンだけにインストールされます。 <p>どちらの場合も、パッケージがインストールされたすべてのゾーンで、パッケージの内容全体が可視になります。</p>

表 C-3 ゾーンのパッケージパラメータの有効な設定 (続き)

SUNW_PKG_ALLZONES の設定	SUNW_PKG_HOLLOW の設定	SUNW_PKG_THISZONE の設定	パッケージの説明
false	false	true	<p>この設定を持つパッケージは、大域ゾーンまたは非大域ゾーンにインストールできます。インストール後に新しい非大域ゾーンを作成した場合、パッケージはこれらの新しい非大域ゾーンには伝達されません。</p> <ul style="list-style-type: none">■ 大域ゾーン内で <code>pkgadd</code> コマンドを実行すると、パッケージは大域ゾーンだけにインストールされます。■ 非大域ゾーン内で <code>pkgadd</code> コマンドを実行すると、パッケージはその非大域ゾーンだけにインストールされます。 <p>どちらの場合も、パッケージがインストールされたゾーンで、パッケージの内容全体が可視になります。</p>
true	false	false	<p>この設定を持つパッケージは、大域ゾーンだけにインストールできます。 <code>pkgadd</code> コマンドを実行すると、パッケージは大域ゾーンおよびすべての非大域ゾーンにインストールされます。すべてのゾーンで、パッケージの内容全体が可視になります。</p> <p>注-パッケージを非大域ゾーンにインストールしようすると失敗します。</p>

表 C-3 ゾーンのパッケージパラメータの有効な設定 (続き)

SUNW_PKG_ALLZONES の設定	SUNW_PKG_HOLLOW の設定	SUNW_PKG_THISZONE の設定	パッケージの説明
true	true	false	<p>この設定を持つパッケージは、大域管理者が大域ゾーンだけにインストールできます。pkgadd コマンドを実行すると、パッケージの内容が大域ゾーンに完全にインストールされます。パッケージパラメータの値がこのように設定されている場合、パッケージの内容自体はどの非大域ゾーンにも提供されません。パッケージをインストール済みとして表示するために必要なパッケージインストール情報だけが、すべての非大域ゾーンにインストールされます。これにより、このパッケージに依存するほかのパッケージをインストールできるようになります。</p> <p>「hollow」パッケージの詳細については、『Oracle Solaris の管理: Oracle Solaris コンテナ - リソース管理と Oracle Solaris ゾーン』の第 25 章「ゾーンがインストールされている Oracle Solaris システムでのパッケージとパッチについて (概要)」を参照してください。</p> <p>パッケージの依存関係を検査できるように、パッケージはすべてのゾーンでインストール済みとして表示されます。</p> <ul style="list-style-type: none"> ■ 大域ゾーンでは、パッケージの内容全体が可視になります。 ■ 完全ルート非大域ゾーンでは、パッケージの内容全体が不可視になります。 ■ 非大域ゾーンが大域ゾーンからファイルシステムを継承する場合、このファイルシステムにインストールされているパッケージは非大域ゾーンで可視になります。パッケージで提供されるほかのすべてのファイルは、非大域ゾーン内では不可視になります。 <p>たとえば、疎ルート非大域ゾーンは、特定のディレクトリを大域ゾーンと共有します。これらのディレクトリは読み取り専用です。疎ルート非大域ゾーンは、/platform ファイルシステムをほかのゾーンと共有します。もう 1 つの例は、ブートするハードウェアだけに関連するファイルがパッケージで提供されている場合です。</p> <p>注-パッケージを非大域ゾーンにインストールしようとすると失敗します。</p>

説明	参照先
パッケージとゾーンの詳細	『Oracle Solaris の管理: Oracle Solaris コンテナ - リソース管理と Oracle Solaris ゾーン』の第 25 章「ゾーンがインストールされている Oracle Solaris システムでのパッケージとパッチについて (概要)」
疎ルートゾーンと完全ルートゾーンの概要	『Oracle Solaris の管理: Oracle Solaris コンテナ - リソース管理と Oracle Solaris ゾーン』の第 16 章「Solaris ゾーンの紹介」
パッケージの特性およびパラメータの詳細	<code>pkginfo(4)</code> のマニュアルページ
パッケージパラメータ値の表示の詳細	<code>pkgparam(1)</code> のマニュアルページ

詳細情報

次のレファレンスでは、この付録のトピックに関する背景情報について説明しています。

パッケージの要件の詳細および用語の定義	『アプリケーションパッケージ開発者ガイド』の第 6 章「パッケージの作成のための高度な手法」
パッケージの追加と削除およびインストール管理ファイルに関する基本情報	『Oracle Solaris の管理: 基本管理』の第 20 章「ソフトウェアの管理 (概要)」
この付録に記載されている個々のコマンドの詳細について参照するマニュアルページ	<code>dircmp(1)</code> 、 <code>fssnap(1M)</code> 、 <code>ps(1)</code> 、または <code>truss(1)</code> <code>pkgadd(1M)</code> 、 <code>pkgchk(1M)</code> 、または <code>pkgrm(1M)</code> のマニュアルページ
Live Upgrade の概要	第 2 章「Live Upgrade (概要)」
JumpStart の概要	『Oracle Solaris 10 1/13 インストールガイド: JumpStart インストール』の第 2 章「JumpStart (概要)」
Oracle Solaris ゾーンの概要	『Oracle Solaris の管理: Oracle Solaris コンテナ - リソース管理と Oracle Solaris ゾーン』の第 16 章「Solaris ゾーンの紹介」

アップグレード時のパッチアナライザの使用(タスク)

この章では、Oracle Solaris OS をアップグレードする前にパッチアナライザでパッチを検査する手順について説明します。次のいずれかにアップグレードする場合は、パッチアナライザがシステムの解析を実行します。

- Solaris 10 1/06 リリース
- Solaris 10 6/06 リリース
- Solaris 10 11/06
- Solaris 10 8/07
- Solaris 10 5/08
- Solaris 10 10/08

Oracle Solaris Update リリースへのアップグレード

すでに Solaris 10 3/05 リリース OS を実行していて、個別のパッチをインストール済みの場合、以降の Oracle Solaris 10 リリースにアップグレードしたときの動作は次のとおりです。

- 上記のリリースの1つでその一部として提供されているすべてのパッチが、システムに再び適用されます。これらのパッチはバックアウトできません。
- システムにすでにインストールされていたパッチのうち、上記のリリースの1つに含まれていないものは、すべて削除されます。

パッチアナライザは、システムを解析し、上記のいずれかのリリースへのアップグレードによって削除されるパッチがあれば、それを特定します。パッチアナライザは次の形式で使用できます。

- Oracle Solaris のインストールプログラムを使用してアップグレードを行なっている場合は、「パッチの解析」ダイアログボックスが表示されます。「はい」を選択して解析を実行します。
- テキストインストーラを使用してアップグレードする場合は、「パッチの解析」ダイアログボックスの「解析」を選択して解析を実行します。

- JumpStart インストールまたは Oracle Solaris の機能である Live Upgrade を使用してアップグレードする場合は、`analyze_patches` スクリプトを実行して解析を実行します。

▼ `analyze_patches` を使用する方法

注-`analyze_patches` スクリプトを実行するには、インストール済みシステムと Oracle Solaris Operating System DVD、Oracle Solaris Software CD、またはネットワークインストールイメージに、NFS またはローカルでマウントされたメディアを通してスクリプトからアクセスできることが必要です。

1 Misc ディレクトリに移動します。

この例では、イメージはローカルにマウントされたメディア上にあります。

```
# cd /cdrom/cdrom0/Solaris_10/Misc
```

2 `analyze_patches` スクリプトを実行します。

```
# ./analyze_patches -R rootdir -N netdir -D databasedir
```

-R *rootdir* *rootdir* は、インストールされたシステムのルートです。デフォルトは `/` です。

-N *netdir* *netdir* は、インストールする OS イメージのルートへのパスです。デフォルトは `/cdrom/cdrom0` です。*netdir* は、`Solaris_10` ディレクトリを含んでいるディレクトリのパスです。NFS マウントポイントから `patch_analyzer` を実行している場合は、このオプションを使用する必要があります。

-D *databasedir* OS イメージ内の `Misc/` ディレクトリ以外のディレクトリからスクリプトを呼び出すと、プログラムはパッチの解析に使用するデータベースを見つけることができません。-D オプションを使用して、データベースへのパスを指定してください。このデータベースは OS イメージの `Solaris_10 /Misc/database` に置かれています。このデータベースを使用しなければ、スクリプトは正しく動作しません。

3 パッチアナライザの出力を確認します。

パッチアナライザでは、ほかのパッチによって削除、ダウングレード、累積、または廃止されるパッチのリストが得られます。パッチの累積はパッチのアップグレードに似ています。累積されるパッチは削除され、その修正は新しいパッチによって提供されます。次のようなメッセージが表示されます。

```
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.
```

パッチアナライザプログラムでリストが得られない場合は、システムにすでにインストールされていたどのパッチに対しても何のアクションも行われません。

- 4 パッチの置換と削除に問題がない場合は、システムをアップグレードします。

用語集

DHCP	Dynamic Host Configuration Protocol (動的ホスト構成プロトコル) の略。アプリケーション層のプロトコル。TCP/IP ネットワーク上の個々のコンピュータつまりクライアントが、中央管理を行なっている指定の DHCP サーバーから IP アドレスなどのネットワーク構成情報を抽出できるようにします。この機能は、大規模な IP ネットワークの保持、管理によるオーバーヘッドを削減します。
/etc ディレクトリ	重要なシステム構成ファイルや保守コマンドが収められているディレクトリ。
/export ファイルシステム	OS サーバー上のファイルシステムで、ネットワーク上のほかのシステムと共有されます。たとえば、 /export ファイルシステムには、ディスクレスクライアント用のルート (/) ファイルシステムとスワップ空間、それにネットワーク上のユーザーのホームディレクトリを収めることができます。ディスクレスクライアントは、ブートと実行の際に OS サーバー上の /export ファイルシステムに依存します。
fdisk パーティション	x86 ベースのシステム上にある特定のオペレーティングシステム専用のディスクドライブの論理パーティション。Oracle Solaris ソフトウェアをインストールするには、x86 ベースシステム上に 1 つ以上の Solaris fdisk パーティションを設定する必要があります。x86 ベースシステムでは、1 台のディスクに最大 4 つの fdisk パーティションを作成できます。これらのパーティションは、個別のオペレーティングシステムをインストールして使用できます。各オペレーティングシステムは、独自の fdisk パーティション上に存在する必要があります。システムでは、ディスクあたり 1 つの Oracle Solaris fdisk パーティションのみ持つことができます。
GRUB	x86 のみ: GRUB (GNU GRand Unified Bootloader) は、簡単なメニューインタフェースを備えたオープンソースのブートローダーです。メニューには、システムにインストールされているオペレーティングシステムのリストが表示されます。GRUB を使用すると、Oracle Solaris OS、Linux、Microsoft Windows などの各種オペレーティングシステムを簡単にブートすることができます。
GRUB 編集メニュー	x86 のみ: GRUB メインメニューのサブメニューであるブートメニュー。このメニューには、GRUB コマンドが表示されます。これらのコマンドを編集して、ブート動作を変更できます。
GRUB メインメニュー	x86 のみ: システムにインストールされているオペレーティングシステムがリストされたブートメニュー。このメニューから、BIOS または fdisk パーティションの設定を変更することなく、簡単にオペレーティングシステムをブートできます。

JumpStart	ユーザーが定義するプロファイルに基づいて、Oracle Solaris ソフトウェアがシステムに自動的にインストールされるインストール方法。ユーザーやシステムの種類ごとに、カスタマイズされたプロファイルを作成できます。カスタム JumpStart インストールは、ユーザーが作成する JumpStart インストールです。
JumpStart インストール	出荷時にインストールされている JumpStart ソフトウェアを使用することによって、Oracle Solaris ソフトウェアがシステムに自動的にインストールされるインストール方法。
Live Upgrade	アクティブブート環境が稼動している間に複製ブート環境のアップグレードを行うことにより、稼動中の環境のダウンタイムをなくすことを可能にするアップグレード方法。
menu.lst ファイル	x86 のみ: システムにインストールされているすべてのオペレーティングシステムがリストされたファイル。このファイルの内容は、GRUB メニューに表示されるオペレーティングシステムの一覧を記述したものです。GRUB のメニューから、BIOS または fdisk パーティションの設定を変更することなく、簡単にオペレーティングシステムをブートできます。
NIS	SunOS 4.0 (以上) のネットワーク情報サービス。ネットワーク上のシステムとユーザーに関する重要な情報が収められている分散型ネットワークデータベース。NIS データベースは、マスターサーバーとすべてのスレーブサーバーに格納されています。
NIS+	SunOS 5.0 (以上) のネットワーク情報サービス。NIS+ は、SunOS 4.0 (以上) のネットワーク情報サービスである NIS に代わるものです。
/opt ファイルシステム	Sun 以外のソフトウェア製品や別製品のソフトウェア用のマウントポイントが収められているファイルシステム。
Oracle Solaris DVD または CD イメージ	システムにインストールされる Oracle Solaris ソフトウェア。Oracle Solaris DVD または CD 上でアクセスしたり、Oracle Solaris DVD または CD イメージをコピーしたインストールサーバーのハードディスク上でアクセスしたりできます。
Oracle Solaris インストールプログラム	グラフィカルユーザーインターフェース (GUI) とコマンド行インターフェース (CLI) を備えたインストールプログラム。ウィザードパネルを使用して、Oracle Solaris ソフトウェアやサードパーティーソフトウェアをインストールする手順を示します。
Oracle Solaris ゾーン	ソフトウェアによるパーティション分割技術。オペレーティングシステムのサービスを仮想化し、隔離された安全なアプリケーション実行環境を提供します。非大域ゾーンを作成すると、そのアプリケーション実行環境で実行されるプロセスは、ほかのゾーンとは隔離されます。このように隔離することで、あるゾーンで実行中のプロセスが、ほかのゾーンで実行中のプロセスを監視したり操作したりすることを防ぐことができます。「大域ゾーン」と「非大域ゾーン」も参照してください。
RAID-0 ボリューム	ストライプ方式または連結方式のボリューム。これらのコンポーネントはサブミラーとも呼ばれます。ストライプや連結は、ミラーを構築する基本構成ブロックです。
RAID-1 ボリューム	同じデータのコピーを複数保持しているボリューム。RAID-1 ボリュームは、「サブミラー」と呼ばれる 1 つまたは複数の RAID-0 ボリュームから構成されます。RAID-1 ボリュームは「ミラー」と呼ばれることもあります。

RAID-Z ストレージプール	ZFS ストレージプールとして使用できる複数のディスク上にデータとパリティを格納する仮想デバイス。RAID-Z は RAID-5 に似ています。
rules.ok ファイル	rules ファイルから生成されたファイル。カスタム JumpStart インストールソフトウェアは、rules.ok ファイルを使ってシステムとプロファイルを照合します。rules.ok ファイルは、check スクリプトを使用して作成してください。
rules ファイル	自動的にインストールするシステムの各グループまたは単一のシステムのルールを含んでいるテキストファイル。各ルールは1つ以上のシステム属性に基づいてシステムグループを識別します。rules ファイルは、各グループをプロファイル (Oracle Solaris ソフトウェアをどのようにしてグループ内の各システムにインストールするかを定めたテキストファイル) にリンクします。rules ファイルは、カスタム JumpStart インストールで使用されます。プロファイルも参照してください。
/usr ファイルシステム	スタンドアロンシステムまたはサーバー上のファイルシステム。標準 UNIX プログラムの多くが格納されています。ローカルコピーを保持する代わりに、大きな /usr ファイルシステムをサーバーと共有することにより、システム上で Oracle Solaris ソフトウェアをインストールおよび実行するために必要な全ディスク容量を最小限に抑えることができます。
/var ファイルシステム	システムの存続期間にわたって変更または増大が予想されるシステムファイルが格納されている (スタンドアロンシステム上の) ファイルシステムまたはディレクトリ。これらのファイルには、システムログ、vi ファイル、メールファイル、UUCP ファイルなどがあります。
Volume Manager	DVD-ROM、CD-ROM、およびフロッピーディスク上のデータへのアクセスを管理および実行するためのメカニズムを提供するプログラム。
ZFS	ストレージプールを使用して物理ストレージを管理するファイルシステム。
アーカイブ	<p>マスターシステムからコピーされたファイルの集合体。このファイルには、アーカイブの名前や作成した日付など、アーカイブの識別情報が含まれています。アーカイブをシステムにインストールすると、システムはマスターシステムとまったく同じ構成になります。</p> <p>アーカイブは差分アーカイブ、つまり更新前のマスターイメージと更新されたマスターイメージの相違部分のみを含むフラッシュアーカイブとすることも可能です。差分アーカイブには、クローンシステムで保持、変更、または削除するファイルが含まれます。差分更新により、指定されたファイルだけが更新されます。また、差分更新を使用可能なシステムは、更新前のマスターイメージとの整合性を保持するソフトウェアを含むシステムのみに限定されます。</p>
アップグレード	<p>ファイルを既存のファイルとマージし、可能な場合には変更を保持するインストール。</p> <p>Oracle Solaris OS のアップグレードでは、Oracle Solaris OS の新しいバージョンがシステムのディスク上の既存のファイルにマージされます。アップグレードでは、Oracle Solaris OS の以前のバージョンに対して行なった変更は最大限に保存されます。</p>

アップグレードオプション	Oracle Solaris のインストールによって提示されるオプション。アップグレード手順では、新しいバージョンの Oracle Solaris とディスク上の既存のファイルが結合されます。アップグレードでは、前回 Oracle Solaris をインストールしてから加えられたローカルの変更内容はできるかぎり残されます。
アンマウント	マシンに接続されたディスクまたはネットワーク上のリモートディスク上のディレクトリへのアクセスを解除するプロセス。
インストールサーバー	ネットワーク上のほかのシステムが Oracle Solaris をインストールできるように、Oracle Solaris DVD または CD のイメージの提供元となるサーバー (メディアサーバーとも呼ばれる)。Oracle Solaris DVD または CD のイメージをサーバーのハードディスクにコピーすることによってインストールサーバーを作成できます。
エンドユーザーシステムサポート	コアシステムサポートソフトウェアグループのほかに、エンドユーザーに推奨するソフトウェアが収められているソフトウェアグループ。これには共通デスクトップ環境 (CDE) や DeskSet ソフトウェアが含まれます。
開始スクリプト	ユーザーが定義する Bourne シェルスクリプト。rules ファイル内で指定され、Oracle Solaris ソフトウェアがシステムにインストールされる前にタスクを実行します。開始スクリプトは、JumpStart インストールでのみ使用できます。
開発者システムサポート	エンドユーザーシステムサポートソフトウェアグループのほかに、ソフトウェア開発用のライブラリ、インクルードファイル、マニュアルページ、およびプログラミングツールが収められているソフトウェアグループ。
仮想デバイス	ZFS プール内の論理デバイス。物理デバイス、ファイル、または一連のデバイスを仮想デバイスに設定できます。
クリティカルファイルシステム	Oracle Solaris OS が必要とするファイルシステム。Oracle Solaris の機能である Live Upgrade を使用するとき、これらのファイルシステムは、アクティブブート環境と非アクティブブート環境のそれぞれの <code>vfstab</code> ファイルでは独立したマウントポイントになります。root (/)、/usr、/var、/opt などがファイルシステムの例です。これらのファイルシステムは、必ずソースブート環境から非アクティブブート環境にコピーされます。
限定ネットワークシステムサポート	Oracle Solaris システムのブートおよび実行に必要な最小限のコードが含まれ、ネットワークサービスのサポートが制限されているソフトウェアグループ。限定ネットワークシステムサポートは、テキストベースのマルチユーザーコンソールと、システム管理ユーティリティを提供します。このソフトウェアグループを使用すると、システムでネットワークインタフェースを認識できますが、ネットワークサービスがアクティブになることはありません。
コアシステムサポート	システムで Oracle Solaris OS をブートして実行するために必要な最小限のソフトウェアが収められているソフトウェアグループ。コアには共通デスクトップ環境 (CDE) を実行するために必要ないくつかのネットワーク用ソフトウェアとドライバが含まれます。CDE ソフトウェアは、コアには含まれません。
更新	システムにインストールを実行して同じタイプのソフトウェアを変更することまたはそのインストール自体。アップグレードとは異なり、更新によりシステムがダウングレードされる場合があります。初期インストールとは異なり、更新を実行するには同じタイプのソフトウェアがあらかじめインストールされていなければなりません。

サブミラー	RAID-0 ボリュームを参照してください。
終了スクリプト	ユーザーが定義する Bourne シェルスクリプト。rules ファイル内で指定され、Oracle Solaris ソフトウェアがシステムにインストールされてから、システムがリブートされるまでの間にタスクを実行します。終了スクリプトは、JumpStart インストールで使われます。
状態データベース	Solaris ボリュームマネージャー構成の状態に関する情報を保存するデータベース。状態データベースは、複製された複数のデータベースコピーの集まりです。各コピーは「状態データベースの複製」と呼ばれます。状態データベースは、既知の状態データベースの複製の格納場所とステータスをすべて記録しています。
状態データベースの複製	状態データベースのコピー。複製により、データベース内のデータの有効性が保証されます。
初期インストール	現在実行中のソフトウェアを上書きするか、空のディスクを初期化するインストール。 Oracle Solaris OS の初期インストールでは、システムのディスクが Oracle Solaris OS の新しいバージョンで上書きされます。システム上で Oracle Solaris OS が稼働していない場合は、初期インストールを行う必要があります。アップグレード可能な Oracle Solaris OS がシステム上で稼働している場合は、初期インストールによってディスクが上書きされ、OS やローカルの変更は保持されません。
スライス	ソフトウェアごとに分割される、ディスク領域の区分。
スワップ空間	メモリーに再ロードできる状態になるまで、メモリー領域の内容を一時的に保持するスライスまたはファイル。/swap または swap ボリュームとも呼ばれます。
全体ディストリビューション	Oracle Solaris リリース全体が含まれているソフトウェアグループ。
全体ディストリビューションと OEM サポート	Oracle Solaris リリース全体と、OEM のための追加ハードウェアサポートを含むソフトウェアグループ。Oracle Solaris ソフトウェアを SPARC ベースサーバーにインストールする場合は、このソフトウェアグループを推奨します。
ゾーン	非大域ゾーンを参照してください
ソフトウェアグループ	Oracle Solaris ソフトウェアの論理グループ(クラスとパッケージ)。Oracle Solaris のインストール中は、次のいずれかのソフトウェアグループをインストールできます。コアシステムサポート、エンドユーザーシステムサポート、開発者システムサポート、または全体ディストリビューションです。また、SPARC システムのみ、全体ディストリビューションと OEM サポートもインストールできます。
大域ゾーン	Oracle Solaris ゾーンでは、大域ゾーンはシステムのデフォルトのゾーンであり、システム全体の管理に使用されるゾーンでもあります。非大域ゾーンの構成、インストール、管理、およびアンインストールは、大域ゾーンからのみ行うことができます。物理デバイス、ルーティング、動的再構成 (DR) といったシステムインフラストラクチャーの管理は、大域ゾーンでのみ行うことができます。大域ゾーンで実行されるプロセスは、適切な権限が付与されていれば、ほかのゾーンに関連付けられているオブジェクトにもアクセスできます。Oracle Solaris ゾーンと非大域ゾーンも参照してください。

データセット	次の ZFS エンティティの総称名。クローン、ファイルシステム、スナップショット、またはボリューム。
ネットワークインストール	CD-ROM または DVD-ROM ドライブがあるシステムから CD-ROM または DVD-ROM ドライブがないシステムにネットワークを介してソフトウェアをインストールする方法。ネットワークインストールを行うには、「ネームサーバー」と「インストールサーバー」が必要です。
ネットワークに接続されたシステム	ハードウェアやソフトウェアを介して接続されているシステムのグループ (ホスト)。通信や情報の共有が可能です。ローカルエリアネットワーク (LAN) とも呼ばれます。システムをネットワークに接続するには、通常、1 台以上のサーバーが必要です。
ネットワークに接続されていないシステム	ネットワークに接続されていない、またはほかのシステムに依存しないシステム。
パッケージ	モジュール形式でのインストールを可能にするソフトウェアの集まり。Oracle Solaris ソフトウェアはソフトウェアグループに分割され、それぞれがクラスとパッケージから構成されています。
パッチアナライザ	手作業で実行したり、Oracle Solaris インストールプログラムの一部として実行したりできるスクリプト。パッチアナライザは、システムを解析し、Oracle Solaris Update にアップグレードすることで削除されるパッチがどれであるかを判断します。
非大域ゾーン	Oracle Solaris オペレーティングシステムの単一インスタンス内に作成された仮想オペレーティングシステム環境。非大域ゾーンでは、システムのほかの部分と相互に作用することなく、1 つ以上のアプリケーションを実行できます。非大域ゾーンはゾーンとも呼ばれます。Oracle Solaris ゾーンと大域ゾーンも参照してください。
ファイルサーバー	ネットワーク上のシステムに対して、ソフトウェアやファイルの記憶領域を提供するサーバー。
ファイルシステム	Oracle Solaris OS において、ユーザーがアクセスできるファイルおよびディレクトリから成るツリー構造のネットワーク。
ブート	メモリーにシステムソフトウェアを読み込んで起動すること。
ブートアーカイブ	<p>x86 のみ: ブートアーカイブは、Oracle Solaris OS のブートに使用されるクリティカルなファイルの集まりです。これらのファイルは、ルート (/) ファイルシステムがマウントされる前、システムの起動中に必要です。システムは、2 つのブートアーカイブを維持管理しています。</p> <ul style="list-style-type: none"> ■ システムで Oracle Solaris OS をブートするために使用されるブートアーカイブ。このブートアーカイブは、プライマリブートアーカイブと呼ばれることもあります。 ■ プライマリブートアーカイブが損傷を受けたとき、回復のために使用されるブートアーカイブ。このブートアーカイブは、ルート (/) ファイルシステムをマウントすることなくシステムをブートします。GRUB メニューでは、このブートアーカイブはフェイルセーフと呼ばれます。アーカイブの重要な目的はプライマリブートアーカイブを再生成することであり、通常、プライマリブートアーカイブがシステムのブートに使用されます。

ブート環境	<p>Oracle Solaris OS を操作する上で重要な必須ファイルシステム (ディスクスライスおよびマウントポイント) の集まり。ディスクスライスは、同じ 1 つのディスク上に存在することも、分散された複数のディスク上に存在することもあります。</p> <p>アクティブなブート環境とは、現在ブートしている環境を指します。単一のアクティブなブート環境からだけブートできます。アクティブでないブート環境とは、現在ブートしていないが、次のリブート時にアクティブ化できる状態にある環境のことを指します。</p>
ブートサーバー	<p>同じネットワークのサブネット上のクライアントシステムに、起動に必要なプログラムと情報を提供するサーバーシステム。インストールサーバーが存在するサブネットワークが、Oracle Solaris ソフトウェアをインストールするシステムと異なる場合、ネットワークを介してインストールするにはブートサーバーが必要です。</p>
ブートローダー	<p>x86 のみ: ブートローダーは、システムの電源を入れた後に最初に実行されるソフトウェアプログラムです。このプログラムがブートプロセスを開始します。</p>
プール	<p>デバイスの論理グループ。使用可能な ZFS ストレージのレイアウトおよび物理特性を記述します。データセットの領域は、プールから割り当てられます。</p>
フェイルセーフブートアーカイブ	<p>x86 のみ: プライマリブートアーカイブが損傷を受けたとき、回復のために使用されるブートアーカイブ。このブートアーカイブは、ルート (/) ファイルシステムをマウントすることなくシステムをブートします。このブートアーカイブは、GRUB メニューではフェイルセーフと呼ばれます。アーカイブの重要な目的はプライマリブートアーカイブを再生成することであり、通常、プライマリブートアーカイブがシステムのブートに使用されます。ブートアーカイブを参照してください。</p>
フォーマット	<p>データを一定の構造にしたり、データを保存できるようにディスクをセクターに分割したりすること。</p>
フォールバック	<p>以前に動作していた環境に戻すこと。ブート環境のアクティブ化の処理中や、ブート対象として指定されたブート環境に問題または望ましくない動作が発生する場合にはフォールバックを行います。</p>
プライマリブートアーカイブ	<p>システムで Oracle Solaris OS をブートするために使用されるブートアーカイブ。このブートアーカイブは、プライマリブートアーカイブと呼ばれることもあります。ブートアーカイブを参照してください。</p>
フラッシュアーカイブ	<p>マスターシステムと呼ばれるシステムからファイルのアーカイブを作成できる Oracle Solaris インストール機能。このアーカイブを使ってほかのシステムのインストールを行うと、そのシステムの構成はマスターシステムと同じになります。アーカイブも参照してください。</p>
プロファイル	<p>カスタム JumpStart 方法を使用する場合に、Oracle Solaris ソフトウェアをインストールする方法を定義するテキストファイル。たとえば、プロファイルでインストールするソフトウェアグループを定義します。各ルールは、そのルールが一致したときにシステムがインストールされる方法を定義してあるプロファイルを指定します。通常は、ルールごとに異なるプロファイルを作成します。しかし、複数のルールで同じプロファイルを使用することも可能です。「rules ファイル」も参照してください。</p>

ボリューム	<p>システムで単一の論理デバイスとして扱われる、物理スライスやボリュームの集まり。ボリュームは、アプリケーションやファイルシステムにとって物理ディスクと同じように機能します。</p> <p>一部のコマンド行ユーティリティーでは、ボリュームはメタデバイスと呼ばれます。一般的な UNIX 用語では、ボリュームは、「擬似デバイス」または「仮想デバイス」とも呼ばれます。</p>
マウント	<p>マウント要求を行うマシンのディスクまたはネットワーク上のリモートディスクから、ディレクトリにアクセスするプロセス。ファイルシステムをマウントするには、ローカルシステム上のマウントポイントと、マウントするファイルシステム名(たとえば <code>/usr</code>)が必要です。</p>
マウントポイント	<p>リモートマシン上に存在するファイルシステムのマウント先となる、ワークステーション上のディレクトリ。</p>
ミニルート	<p>Oracle Solaris インストールメディアに収録されている、ブート可能な最小のルート (<code>/</code>) ファイルシステム。ミニルートは、システムのインストールおよびアップグレードに必要な Oracle Solaris ソフトウェアで構成されます。x86 システムでは、ミニルートはシステムにコピーされて、フェイルセーフブートアーカイブとして使用されます。「フェイルセーフブートアーカイブ」を参照してください。</p>
ミラー	<p>RAID-1 ボリュームを参照してください。</p>
ルート	<p>複数の項目から成る階層構造の最上位。ルートは、ほかのすべての項目を子孫として持つ唯一の項目です。「ルートディレクトリ」または「ルート (<code>/</code>) ファイルシステム」を参照してください。</p>
ルート (<code>/</code>) ファイルシステム	<p>ほかのすべてのファイルシステムの元となる最上位ファイルシステム。ルート (<code>/</code>) ファイルシステムはほかのすべてのファイルシステムがマウントされる元となり、マウント解除されることはありません。ルート (<code>/</code>) ファイルシステムには、カーネル、デバイスドライバ、システムのブート (ブート) に使用されるプログラムなど、システムの稼働に不可欠なディレクトリやファイルが含まれています。</p>
ルートディレクトリ	<p>ほかのすべてのディレクトリの元となる最上位ディレクトリ。</p>
ルール	<p>1 つ以上のシステム属性をプロファイルに割り当てる一連の値。ルールは、カスタム JumpStart インストールで使用されます。</p>
ロケール	<p>同一の言語、風俗、慣習、文化などを共有する地理上または政治上の地域圏 (コミュニティ)。たとえば、米国英語のロケールは <code>en_US</code>、英国英語のロケールは <code>en_UK</code> です。</p>

索引

B

boot: cannot open /kernel/unix メッセージ, 228
bootparams ファイル, 更新, 234

C

Can't boot from file/device メッセージ, 228
CHANGE DEFAULT BOOT DEVICE
 メッセージ, 235
CLIENT MAC ADDR エラーメッセージ, 233
clock gained xxx days メッセージ, 228

E

/etc/bootparams ファイル, JumpStart ディレクトリ
 アクセスの有効化, 234

L

le0: No carrier - transceiver cable problem
 メッセージ, 228
Live Upgrade ジョブの取り消し, 134
Live Upgrade のコマンド, 225
lumount コマンド, 非大域ゾーン, 158

N

No carrier - transceiver cable problem
 メッセージ, 228

Not a UFS filesystem メッセージ, 228

R

RAID-0 ボリューム、説明, 29
RAID-1 ボリューム (ミラー)
 Solaris Volume Manager ボリュームへの移行
 例, 172
 作成とアップグレードの例, 168
 作成の例, 85, 86, 87
 説明, 27, 29
 要件, 51
RAID-1 ボリューム (ミラー) 用スライスの選
 択, 51
RPC Timed out メッセージ, 233

S

Solaris Volume Manager
 Live Upgrade で使用されるコマンド, 52
 例
 RAID-1 ボリュームへの移行, 172
Solaris ボリュームマネージャー
 例
 RAID-1 ボリュームの切り離しとアップグ
 レード, 168

T

timed out RPC エラー, 233

transceiver cable problem メッセージ, 228

U

UFS から ZFS への移行, 195

非大域ゾーンがある場合, 215

Unknown client error メッセージ, 227

V

Volume Manager, 「Solaris Volume Manager」を参照

W

WARNING: CHANGE DEFAULT BOOT
DEVICE, 235

WARNING: clock gained xxx days メッセージ, 228

Z

ZFS

UFS から ZFS への移行, 195

移行

非大域ゾーンがある場合, 215

概要, 177

計画, 189

ブート環境の作成, 195

新しいプールでの, 207

同一プール内, 202

別のソースから, 212

あ

アーカイブ

インストールの例, 36

空のブート環境の作成, 81

ブート環境へのインストール, 108

アップグレード

Oracle Solaris Update リリースへ, 257-259

アップグレード (続き)

アップグレードの失敗, 239

アップグレードの障害回復, 121

ガイドライン, 92

説明, 34

タスク, 92

フラッシュアーカイブのインストール, 108

タスクマップ, 91-92

非大域ゾーン

lumount コマンド, 158

概要, 142

詳細な手順, 149

図, 142

独立したファイルシステム, 148

ファイルシステムの一覧の表示, 156

ブート環境の管理, 156

ブート環境の比較, 157

例, 154

ブート環境

ブート環境, 92

例, 159, 168, 172

アップグレードの失敗, リブートの問題, 239

アップグレードの障害, 回復, 121

い

インストール

パッケージ, 62

プロファイルによるフラッシュアーカイブ
の, 112

フラッシュアーカイブ, 108

か

概要, 19

ZFS, 177

グラフィック, 20

き

キーワード

プロファイル, 101

キーワード (続き)

ボリューム, 84
共有可能ファイルシステム, 定義, 22

く

クリティカルファイルシステム, 定義, 22

け

計画, 43
ZFS, 189

こ

コピー, ファイルシステム, 133

さ

削除, ブート環境, 135
作成
RAID-1 ボリューム (ミラー), 説明, 27
タスク
ZFS, 195
タスクマップ, 61-62
ブート環境, 説明, 22
ブート環境, タスク, 66, 70, 71
サブミラー, 説明, 29

し

状態データベース, 説明, 29

す

ステータス, ブート環境の表示, 132
スライス, 選択するための指針, 50
スワップファイルシステム, スライスを選択する
ための指針, 53

て

ディスク容量の要件, 46-47
テスト, プロファイル, 103

と

トークンリングカード, ブートエラー, 232
トラブルシューティング
DHCP によるネットワークブート, 233
インストールの一般的な問題
DHCP によるネットワークブート, 233
システムのブート, 234
間違ったサーバーからのブート, 234

な

内容のカスタマイズ, 54

は

パッケージ
Live Upgrade の要件, 247
カスタム JumpStart を使用するときの要件, 247
追加, 48, 93
パッチ
追加, 48, 93
パッチレベルの確認, 44, 65
パッチアナライザ, 257-259

ひ

非大域ゾーン
lumount コマンド, 158
UFS から ZFS への移行, 215
アップグレード例, 154
概要, 142
詳細な手順, 149
図, 142
独立したファイルシステム, 148
ファイルシステムの一覧の表示, 156
ファイルシステムの比較, 157

非大域ゾーン (続き)

- ブート環境の管理, 156

- 必要なパッケージ, 45

表示

- ブート環境の構成, 139

- ブート環境の名前, 136

ふ

- ファイルシステムの比較, 非大域ゾーン, 157

- ファイルとファイルシステム

- RAID-1 ボリューム (ミラー) の作成, 説明, 27

- サイズの見積もり, 47

- 作成のためのガイドライン, 49

- スライスを選択するための指針, 50

- 説明, 22

- ブート環境間でのファイルシステムの共有, 54

- ブート環境, 失敗、説明, 41

- ブート環境のアクティブ化

- 失敗、説明, 41

- 説明, 39

- タスク, 113

- ファイルの同期化、説明, 55

- ブート環境の構成の表示, 非大域ゾーン, 156

- ブート環境の名前の変更, 137

- ブート環境の比較, 135

- プロファイル

- 差分アーカイブ用の例, 102

- 例, 101

- プロファイルキーワード

- forced_deployment

- 説明と値, 107

- local_customization

- 説明と値, 108

ほ

- ボリューム

- RAID-0、説明, 29

- RAID-1、説明, 29

み

- ミラー、「RAID-1 ボリューム」を参照

よ

- 要件, Live Upgrade の使用, 43

る

- ルート (/) ファイルシステム, 非アクティブブート環境に対するパッケージ要件, 247

れ

- 例, 159

- RAID-1 ボリュームのアップグレード, 168, 172

- RAID-1 ボリュームの作成, 85, 86

- 完全なプロセス

- 完全なプロセス, 159

- ミラーの作成, 87

- 連結, 説明, 29