



Solaris 9 インストールガイド

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

Part No: 816-3905-10
2002 年 5 月

Copyright 2002 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

本製品およびそれに関連する文書は著作権法により保護されており、その使用、複製、頒布および逆コンパイルを制限するライセンスのもとにおいて頒布されます。サン・マイクロシステムズ株式会社の書面による事前の許可なく、本製品および関連する文書のいかなる部分も、いかなる方法によっても複製することが禁じられます。

本製品の一部は、カリフォルニア大学からライセンスされている Berkeley BSD システムに基づいていることがあります。UNIX は、X/Open Company, Ltd. が独占的にライセンスしている米国ならびに他の国における登録商標です。フォント技術を含む第三者のソフトウェアは、著作権により保護されており、提供者からライセンスを受けているものです。

Federal Acquisitions: Commercial Software—Government Users Subject to Standard License Terms and Conditions.

本製品に含まれる HG 明朝 L、HG-MincyoL-Sun、HG ゴシック B、および HG-GothicB-Sun は、株式会社リコーがリコービマジクス株式会社からライセンス供与されたタイプフェイスマスタをもとに作成されたものです。HG 平成明朝体 W3@X12 は、株式会社リコーが財団法人日本規格協会からライセンス供与されたタイプフェイスマスタをもとに作成されたものです。フォントとして無断複製することは禁止されています。

Sun、Sun Microsystems、docs.sun.com、AnswerBook、AnswerBook2、JumpStart、Solaris Web Start、Power Management は、米国およびその他の国における米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします) の商標もしくは登録商標です。

サンロゴマークおよび Solaris は、米国 Sun Microsystems 社の登録商標です。

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャに基づくものです。

OPENLOOK、OpenBoot、JLE は、サン・マイクロシステムズ株式会社の登録商標です。

Wnn は、京都大学、株式会社アステック、オムロン株式会社で共同開発されたソフトウェアです。

Wnn6 は、オムロン株式会社、オムロンソフトウェア株式会社で共同開発されたソフトウェアです。© Copyright OMRON Co., Ltd. 1995-2000. All Rights Reserved. © Copyright OMRON SOFTWARE Co., Ltd. 1995-2002 All Rights Reserved.

「ATOK」は、株式会社ジャストシステムの登録商標です。

「ATOK Server/ATOK12」は、株式会社ジャストシステムの著作物であり、「ATOK Server/ATOK12」にかかる著作権その他の権利は、株式会社ジャストシステムおよび各権利者に帰属します。

本製品に含まれる郵便番号辞書 (7 桁/5 桁) は郵政事業庁が公開したデータを元に制作された物です (一部データの加工を行なっています)。

本製品に含まれるフェイスマーク辞書は、株式会社ビレッジセンターの許諾のもと、同社が発行する『インターネット・パソコン通信フェイスマークガイド '98』に添付のものを使用しています。© 1997 ビレッジセンター

Unicode は、Unicode, Inc. の商標です。

本書で参照されている製品やサービスに関しては、該当する会社または組織に直接お問い合わせください。

OPEN LOOK および Sun Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカル・ユーザインタフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

DtComboBox ウィジェットと DtSpinBox ウィジェットのプログラムおよびドキュメントは、Interleaf, Inc. から提供されたものです。(© 1993 Interleaf, Inc.)

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われぬものとします。

本製品が、外国為替および外国貿易管理法 (外為法) に定められる戦略物資等 (貨物または役務) に該当する場合、本製品を輸出または日本国外へ持ち出す際には、サン・マイクロシステムズ株式会社の事前の書面による承諾を得ることのほか、外為法および関連法規に基づく輸出手続き、また場合によっては、米国商務省または米国所轄官庁の許可を得ることが必要です。

原典: Solaris 9 Installation Guide

Part No: 806-5205-10

Revision A



020327@2851



目次

はじめに	15
1 Solaris のインストールおよびアップグレードの計画	19
2 Solaris のインストールおよびアップグレードの計画の概要	21
作業マップ: Solaris ソフトウェアのインストールまたはアップグレード	21
初期インストールかアップグレードか	23
初期インストール	23
アップグレード	23
システム要件	24
メモリー要件	24
Solaris 9 INSTALLATION CD を使用するための要件	25
ネットワークからインストールするか DVD または CD を使ってインストールするか	25
DVD メディアの使用	26
3 Solaris インストール方法の選択	27
Solaris Web Start	27
Solaris suninstall プログラム	28
カスタム JumpStart	28
フラッシュインストール機能	29
Solaris Live Upgrade の方法	30
ファクトリ JumpStart	30

4	Solaris のインストールおよびアップグレードの準備	31
5	ディスク容量の割り当てに関する指針	33
	ディスク容量に関する一般的な計画と推奨事項	33
	ソフトウェアグループごとの推奨ディスク容量	34
6	インストールやアップグレードの前に収集すべき情報	37
	インストール用のチェックリスト	37
	インストール用のワークシート	38
	アップグレード用のチェックリスト	42
	アップグレード用のワークシート	43
7	システム構成情報の事前設定	47
	システム構成情報を事前設定することの利点	47
	システム構成情報の事前設定方法	48
	sysidcfg ファイルによる事前設定	49
	sysidcfg ファイルの構文規則	50
	sysidcfg ファイルキーワード	50
	▼ sysidcfg 構成ファイルを作成する方法	53
	ネームサービスによる事前設定	54
	▼ NIS を使ってロケールを事前設定する方法	54
	▼ NIS+ を使ってロケールを事前設定する方法	56
	電源管理情報の事前設定	57
8	Solaris オペレーティング環境のアップグレード	59
	アップグレード	59
	Solaris Live Upgrade の利用	60
	カスタム JumpStart によるアップグレード	60
	ディスク容量の再配置を使用するアップグレード	61
	アップグレード前のシステムのバックアップ	62
9	ネットワークインストールの準備	63
10	ネットワークインストールの準備 (概要)	65
	ネットワークインストールの計画	65

	ネットワークインストールに必要なサーバー	65
	DHCP によるネットワークインストールパラメータの指定	67
11	DVD メディアを使用してネットワークからインストールするための準備	69
	作業マップ: DVD メディアを使用して Solaris ソフトウェアをネットワークからインストールする準備	69
	DVD メディアを用いたインストールサーバーの作成	70
	▼ DVD メディアを使用してインストールサーバーを作成する方法	71
	DVD イメージを用いたサブネット上でのブートサーバーの作成	74
	▼ DVD イメージを使用してサブネット上にブートサーバーを作成する方法	74
	ネットワークからインストールするシステムの追加	76
	▼ サーバーから <code>add_install_client</code> を実行してネットワークからインストールするシステムを追加する方法	76
12	CD メディアを使用してネットワークからインストールするための準備	79
	作業マップ: CD メディアを使用してネットワーク上でインストールするための準備	80
	CD メディアによるインストールサーバーの作成	81
	▼ CD メディアを使用してインストールサーバーを作成する方法	81
	CD イメージによるサブネット上でのブートサーバーの作成	86
	▼ CD イメージを使用してサブネット上にブートサーバーを作成する方法	86
	ネットワークからインストールするシステムの追加	88
	▼ サーバーから <code>add_install_client</code> を実行してネットワークからインストールするシステムを追加する方法	88
13	ネットワークからインストールするための準備 (コマンドリファレンス)	91
14	Solaris Web Start の使用	93
	Solaris Web Start の実行方法	93
	Solaris Web Start によるインストールまたはアップグレード	94
	作業マップ: Solaris Web Start インストールの実行	94
	▼ Solaris Web Start を使用してインストールまたはアップグレードする方法	95
	Solaris Web Start によるインストールまたはアップグレード後に必要な作業	98
	▼ アップグレード後にローカル変更を復元する方法	98

- 15 **Solaris suninstall プログラムの使用 101**
 - Solaris suninstall プログラム 101
 - Solaris suninstall プログラムによるインストールまたはアップグレード 102
 - 作業マップ: Solaris suninstall プログラムを実行する 102
 - ▼ Solaris suninstall プログラムを使用してインストールまたはアップグレードするには 103
 - Solaris suninstall プログラムによるアップグレード後に必要な作業 104
 - ▼ アップグレード後にローカル変更を復元する方法 104

- 16 **フラッシュインストール機能 105**

- 17 **フラッシュの概要と計画 107**
 - フラッシュの概要 107
 - フラッシュインストールの計画 108
 - マスターシステムへのインストールの設計 108
 - フラッシュアーカイブの作成の計画 110
 - フラッシュアーカイブのインストールの計画 111

- 18 **フラッシュアーカイブの作成 115**
 - 作業マップ: フラッシュアーカイブを作成する作業マップ 115
 - フラッシュアーカイブ作成のための作業 115
 - ▼ マスターシステムにインストールする方法 116
 - ▼ フラッシュアーカイブを作成する方法 116

- 19 **フラッシュアーカイブのインストール 119**
 - Solaris Live Upgrade によるフラッシュアーカイブのインストール 119
 - Solaris Web Start によるフラッシュアーカイブのインストール 120
 - ▼ Solaris Web Start 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 120
 - Solaris suninstall プログラムによるフラッシュアーカイブのインストール 121
 - ▼ Solaris suninstall プログラム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 121
 - カスタム JumpStart インストールによるフラッシュアーカイブのインストール 122
 - ▼ カスタム JumpStart インストー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 123
 - フラッシュのカスタム JumpStart プログラムファイルの例 124

20	フラッシュに関するリファレンス情報	125
	フラッシュアーカイブのセクション	125
	フラッシュのキーワード	126
	一般的なキーワード	126
	identification セクションのキーワード	126
	フラッシュのコマンド	128
	flarcreate	128
	flar	131
21	カスタム JumpStart インストール	135
22	カスタム JumpStart の概要	137
	カスタム JumpStart とは	137
	JumpStart プログラムが Solaris ソフトウェアをインストールする方法	138
23	カスタム JumpStart インストールの準備	143
	作業マップ: カスタム JumpStart インストールの準備	144
	ネットワーク上のシステム用のプロファイルサーバーの作成	145
	▼ JumpStart ディレクトリをサーバー上に作成する方法	146
	▼ 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する方法	147
	スタンドアロンシステム用のプロファイルフロッピーディスクの作成	149
	▼ プロファイルフロッピーディスクを作成する方法	150
	rules ファイルの作成	152
	rules ファイルの構文	152
	▼ rules ファイルを作成する方法	153
	rules ファイルの例	154
	プロファイルの作成	155
	プロファイルの構文	156
	▼ プロファイルを作成する方法	156
	プロファイルの例	157
	プロファイルのテスト	159
	▼ 一時的な Solaris 9 環境を作成してプロファイル进行测试する方法	159
	▼ プロファイル进行测试する方法	160
	プロファイルテストの例	162
	rules ファイルの妥当性を検査する	163
	▼ rules ファイルの妥当性を検査する方法	163

- 24 カスタム **JumpStart** オプション機能の使用 165
 - begin スクリプトの作成 165
 - begin スクリプトに関する注意事項 165
 - begin スクリプトによる動的プロファイルの作成 166
 - finish スクリプトの作成 167
 - finish スクリプトに関する注意事項 167
 - ▼ finish スクリプトでファイルを追加する方法 167
 - finish スクリプトによるパッケージとパッチの追加 168
 - finish スクリプトによるルート環境のカスタマイズ 169
 - finish スクリプトによるシステムのルートパスワードの設定 170
 - finish スクリプトを使用して Web Start インストールプログラムでソフトウェアをインストールする 171
 - 圧縮された構成ファイルの作成 172
 - ▼ 圧縮された構成ファイルを作成する方法 172
 - 圧縮された構成ファイルの例 173
 - ディスク構成ファイルの作成 173
 - ▼ ディスク構成ファイルを作成する方法 173
 - ディスク構成ファイルの例 174
 - サイト固有のインストールプログラムの使用 175

- 25 カスタムルールおよびプローブキーワードの作成 177
 - プローブキーワード 177
 - custom_probes ファイルの作成 178
 - custom_probes ファイルの構文 178
 - custom_probes の関数名の構文 179
 - ▼ custom_probes ファイルを作成する方法 179
 - custom_probes ファイルとキーワードの例 179
 - custom_probes ファイルの妥当性検査 181
 - ▼ custom_probes ファイルを妥当性検査する方法 181

- 26 カスタム **JumpStart** インストールの実行 183
 - カスタム **JumpStart** インストールの実行 183
 - 作業マップ: カスタム **JumpStart** インストールのためのシステムの設定 183
 - ▼ カスタム **JumpStart** プログラムを使用してインストールまたはアップグレードする方法 184

27	カスタム JumpStart によるインストール例	189
	サイトの設定例	189
	インストールサーバーの作成	191
	マーケティングシステム用のブートサーバーの作成	192
	JumpStart ディレクトリの作成	192
	JumpStart ディレクトリの共有	193
	エンジニアリンググループのプロファイルの作成	193
	マーケティンググループのプロファイルの作成	194
	rules ファイルの更新	194
	rules ファイルの妥当性を検査する	195
	ネットワーク上でインストールするためのエンジニアリングシステムの設定	195
	ネットワーク上でインストールするためのマーケティングシステムの設定	196
	システムのブートと Solaris 9 ソフトウェアのインストール	196
28	カスタム JumpStart に関するリファレンス情報	197
	ルールキーワードと値の説明	197
	プロファイルキーワードと値	202
	プロファイルキーワードのリスト	202
	プロファイルキーワードの説明と例	204
	カスタム JumpStart の環境変数	226
	プローブキーワードと値	228
29	Solaris Live Upgrade (トピック)	231
30	Solaris Live Upgrade の概要	233
	Solaris Live Upgrade の紹介	233
	Solaris Live Upgrade の実行手順	234
31	Solaris Live Upgrade の計画	247
	Solaris Live Upgrade のシステム要件	247
	Solaris Live Upgrade のディスク容量の要件	248
	Solaris Live Upgrade によるパッケージおよびバッチの管理	248
	必要なパッケージ	249
	システムパッチレベルのチェック	250
	lucreate コマンドを用いたファイルシステムの作成のための指針	250
	ファイルシステムのスライスを選択するための指針	251

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	251
ルート (/) のミラーとメタデバイスを選択するための指針	252
swap ファイルシステムのスライスを選択するための指針	253
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	253
リモートシステムからの Solaris Live Upgrade の使用	254

32 Solaris Live Upgrade によるブート環境の作成	255
Solaris Live Upgrade インタフェースの概要	255
Solaris Live Upgrade メニューの使用	256
作業マップ: Solaris Live Upgrade のインストールとブート環境の作成	257
Solaris Live Upgrade のインストール	257
▼ Solaris Live Upgrade をインストールする方法	257
Solaris Live Upgrade の起動と停止 (キャラクタインタフェース)	258
▼ Solaris Live Upgrade メニューを起動する	258
▼ Solaris Live Upgrade のメニューを終了するには	259
新しいブート環境の作成	259
▼ ブート環境を作成する (キャラクタインタフェース)	260
▼ ブート環境を初めて作成する (コマンド行インタフェース)	265
▼ ブート環境を作成しファイルシステムをマージする (コマンド行インタフェース)	267
▼ ブート環境を作成しファイルシステムを分割する (コマンド行インタフェース)	269
▼ ブート環境を作成しスワップを再構成する (コマンド行インタフェース)	271
▼ 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 (コマンド行インタフェース)	273
▼ 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 (コマンド行インタフェース)	275
▼ 別々のソースから単一のブート環境を作成 (コマンド行インタフェース)	276
▼ フラッシュアーカイブ用ブート環境の作成 (コマンド行インタフェース)	277
33 Solaris Live Upgrade を用いたアップグレード	279
作業マップ: ブート環境のアップグレード	280
ブート環境のアップグレード	280
メタデバイスやボリュームのアップグレード	281
Solaris Live Upgrade によるパッケージやパッチの管理	282
▼ 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する (キャラクタインタフェース)	283

- ▼ 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する (コマンド行インタフェース) 284
- ▼ 複数の CD を使用してオペレーティングシステムイメージをアップグレードする (コマンド行インタフェース) 284
- ブート環境へのフラッシュアーカイブのインストール 286
 - メタデバイスとボリュームのアップグレード 287
 - ▼ ブート環境へのフラッシュアーカイブのインストール (キャラクタインタフェース) 288
 - ▼ ブート環境へのフラッシュアーカイブのインストール (コマンド行インタフェース) 290
- ブート環境のアクティブ化 291
 - ▼ ブート環境のアクティブ化 (キャラクタインタフェース) 291
 - ▼ ブート環境をアクティブにする (コマンド行インタフェース) 292
 - ▼ ブート環境をアクティブにしてファイルを同期させる (コマンド行インタフェース) 293
- 問題の解決: 元のブート環境へのフォールバック (コマンド行インタフェース) 294
 - ▼ 新しいブート環境の作成に成功した場合のフォールバック 295
 - ▼ 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 295
 - ▼ 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フォールバックする 296

34 Solaris Live Upgrade ブート環境の管理 299

- Solaris Live Upgrade 管理作業の概要 299
- すべてのブート環境のステータスの表示 300
 - ▼ すべてのブート環境のステータスを表示する (キャラクタインタフェース) 301
 - ▼ すべてのブート環境のステータスを表示する (コマンド行インタフェース) 301
- 以前に構成されたブート環境の更新 302
 - ▼ 以前に構成されたブート環境を更新する (キャラクタインタフェース) 302
 - ▼ 以前に構成されたブート環境を更新する (コマンド行インタフェース) 303
- 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 304
 - ▼ スケジュールされた処理 (作成/アップグレード/コピー) を取り消す (キャラクタインタフェース) 304
 - ▼ スケジュールされた処理 (作成/アップグレード/コピー) を取り消す (コマンド行インタフェース) 304
- ブート環境の比較 304
 - ▼ ブート環境を比較する (キャラクタインタフェース) 305
 - ▼ ブート環境を比較する (コマンド行インタフェース) 305
- 非アクティブブート環境の削除 306

▼ 非アクティブブート環境を削除する (キャラクタインタフェース)	306
▼ 非アクティブブート環境を削除する (コマンド行インタフェース)	307
アクティブブート環境の名前の表示	307
▼ アクティブブート環境の名前を表示する (キャラクタインタフェース)	307
▼ アクティブブート環境の名前を表示する (コマンド行インタフェース)	307
ブート環境の名前の変更	308
▼ 非アクティブブート環境の名前を変更する (キャラクタインタフェース)	308
▼ 非アクティブブート環境の名前を変更する (コマンド行インタフェース)	309
ブート環境名に関連付ける説明の作成または変更	309
▼ テキストを使用してブート環境名の説明を作成または変更する方法	309
▼ ファイルを使用してブート環境名の説明を作成または変更する方法	310
▼ テキストで記述された説明からブート環境名を確認する方法	310
▼ ファイル内の説明からブート環境名を確認する方法	311
▼ 名前からブート環境説明を確認する方法	311
ブート環境の構成の表示	312
▼ 非アクティブブート環境の構成を表示する (キャラクタインタフェース)	312
▼ ブート環境の構成を表示する (コマンド行インタフェース)	312
35 Solaris Live Upgradeの使用例	315
Solaris Live Upgrade によるアップグレードの使用例 (コマンド行インタフェース)	315
アクティブブート環境で Live Upgrade をインストールする	315
ブート環境を作成する	316
非アクティブブート環境をアップグレードする	316
ブート環境がブート可能か確認する	317
非アクティブブート環境をアクティブにする	317
ソースブート環境へフォールバックする	317
Solaris Live Upgrade によるアップグレードの例 (キャラクタインタフェース)	318
アクティブブート環境で Live Upgrade をインストールする	318
ブート環境を作成する	319
非アクティブブート環境をアップグレードする	321
非アクティブブート環境をアクティブにする	322

36	Solaris Live Upgrade のコマンドに関するリファレンス情報	323
37	Solaris ソフトウェアに関するリファレンス情報	325
38	Solaris 9 メディアの構成に関するリファレンス情報	327
	Solaris 9 メディア	327
	Solaris 9 メディアのディレクトリ構造	329
	Solaris 9 DVD のディレクトリ構造	329
	Solaris 9 INSTALLATION CD のディレクトリ構造	330
	Solaris 9 SOFTWARE CD のディレクトリ構造	330
	Solaris 9 LANGUAGES CD のディレクトリ構造	332
39	プラットフォーム名とプラットフォームグループに関するリファレンス情報	335
40	ロケールの値 (リファレンス)	337
A	問題発生時の解決方法	345
	ネットワークインストールの設定に関する問題	345
	システムのブートに関する問題	346
	メディアからのブート時のエラーメッセージ	346
	メディアからのブート時の一般的な問題	347
	ネットワークからのブート時のエラーメッセージ	347
	ネットワークからのブート時の一般的な問題	349
	Solaris 9 オペレーティング環境の初期インストール時の問題	350
	Solaris 9 オペレーティング環境のアップグレード時の問題	351
	アップグレード時のエラーメッセージ	351
	アップグレード時の一般的な問題	351
	▼ 問題発生後にアップグレードを継続する方法	352
	Veritas VxVm の実行中に Solaris 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	353
B	リモートからのインストールまたはアップグレード	357
	Solaris Web Start を使用してリモート DVD-ROM または CD-ROM からインストールまたはアップグレードする	357
	▼ リモート DVD-ROM または CD-ROM を使用してインストールまたはアップグレードを行う方法	358

C	その他の SvR4 パッケージ要件に関するリファレンス情報	361
	パッケージ化要件の概要	361
	カスタム JumpStart プログラムと Solaris Live Upgrade の代替ルート (/) 要件	362
	\$PKG_INSTALL_ROOT と \$BASEDIR の相違点の概要	363
	Solaris Live Upgrade 代替ブート環境の要件	364
	カスタム JumpStart プログラムのアップグレード要件	366
D	Solaris Update リリースへのアップグレード	369
	Solaris Update リリースへのアップグレード	369
	▼ analyze_patches スクリプトを実行する方法	370
	▼ パッチアナライザの出力の確認方法	370
	用語集	373
	索引	383

はじめに

本書『Solaris 9 インストールガイド』では、SPARC™ ベースのネットワークに接続されたシステムとネットワークに接続されていないシステムの両方に Solaris™ 9 オペレーティング環境をインストールおよびアップグレードする方法を説明します。

このマニュアルには、システムハードウェアや周辺装置を設定する方法は記載されていません。

対象読者

このマニュアルは、Solaris オペレーティング環境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。このマニュアルには、次の 2 種類の情報が含まれています。

- ネットワーク環境で複数の Solaris マシンを管理するエンタープライズシステム管理者向けの上級 Solaris インストール情報
- Solaris のインストールやアップグレードをときどき行うシステム管理者向けの基本 Solaris インストール情報

関連情報

表 P-1 に、Solaris ソフトウェアをインストールする際に参考となる関連情報の一覧を示します。

表 P-1 関連情報

情報	説明
『Solaris のシステム管理 (基本編)』	システムファイルのバックアップ方法が解説されています。
『Solaris 9 ご使用にあたって (SPARC 版)』	Solaris リリースに関する、バグ、既知の問題、サポートが中止されたソフトウェア、パッチなどが解説されています。
『Solaris 9 Sun ハードウェアマニュアル』	サポート対象のハードウェアについて解説されています。
『Solaris 9 パッケージリスト』	Solaris 9 オペレーティング環境のパッケージの一覧と、その解説です。

Sun のオンラインマニュアル

docs.sun.com では、Sun が提供しているオンラインマニュアルを参照することができます。マニュアルのタイトルや特定の主題などをキーワードとして、検索を行うこともできます。URL は、<http://docs.sun.com> です。

表記上の規則

このマニュアルでは、次のような字体や記号を特別な意味を持つものとして使用します。

表 P-2 表記上の規則

字体または記号	意味	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。 system%
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。	system% su password:
<i>AaBbCc123</i>	変数を示します。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 <i>filename</i> と入力します。

表 P-2 表記上の規則 (続き)

字体または記号	意味	例
『』	参照する書名を示します。	『コードマネージャ・ユーザーズガイド』を参照してください。
「」	参照する章、節、ボタンやメニュー名、強調する単語を示します。	第5章「衝突の回避」を参照してください。 この操作ができるのは、「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅を超える場合に、継続を示します。	<pre>sun% grep `^#define \ XV_VERSION_STRING'</pre>

コード例は次のように表示されます。

■ C シェル

```
machine_name% command y|n [filename]
```

■ C シェルのスーパーユーザー

```
machine_name# command y|n [filename]
```

■ Bourne シェルおよび Korn シェル

```
$ command y|n [filename]
```

■ Bourne シェルおよび Korn シェルのスーパーユーザー

```
# command y|n [filename]
```

[] は省略可能な項目を示します。上記の例は、*filename* は省略してもよいことを示しています。

| は区切り文字 (セパレータ) です。この文字で分割されている引数のうち 1 つだけを指定します。

キーボードのキー名は英文で、頭文字を大文字で示します (例: Shift キーを押します)。ただし、キーボードによっては Enter キーが Return キーの動作をします。

ダッシュ (-) は 2 つのキーを同時に押すことを示します。たとえば、Ctrl-D は Control キーを押したまま D キーを押すことを意味します。

一般規則

- このマニュアルでは、英語環境での画面イメージを使っています。このため、実際に日本語環境で表示される画面イメージとこのマニュアルで使っている画面イメージが異なる場合があります。本文中で画面イメージを説明する場合には、日本語のメニュー、ボタン名などの項目名と英語の項目名が、適宜併記されています。
- このマニュアルでは、「IA」という用語は、Intel 32 ビットのプロセッサアーキテクチャを意味します。これには、Pentium、Pentium Pro、Pentium II、Pentium II Xeon、Celeron、Pentium III、Pentium III Xeon の各プロセッサ、および AMD、Cyrix が提供する互換マイクロプロセッサチップが含まれます。

第 1 章

Solaris のインストールおよびアップグレードの計画

以下の章では、Solaris オペレーティング環境のインストールやアップグレードの計画について説明します。

第 2 章	インストールやアップグレードを行う前に決定すべき項目について説明します。
第 3 章	Solaris をインストールするさまざまな方法について詳しく説明します。インストール環境にとってどの方法が最適であるかを判断する際の参考になります。

第 2 章

Solaris のインストールおよびアップグレードの計画の概要

この章では、Solaris オペレーティング環境のインストールやアップグレードを行う前に決定すべき項目について説明します。この章の内容は次のとおりです。

- 21 ページの「作業マップ: Solaris ソフトウェアのインストールまたはアップグレード」
- 23 ページの「初期インストールかアップグレードか」
- 24 ページの「システム要件」
- 25 ページの「ネットワークからインストールするか DVD または CD を使ってインストールするか」
- 26 ページの「DVD メディアの使用」

注 - このマニュアルでは「スライス」という用語を使用しますが、一部の Solaris のマニュアルとプログラムでは、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。

作業マップ: Solaris ソフトウェアのインストールまたはアップグレード

次の作業マップは、Solaris オペレーティング環境のインストールやアップグレードに必要な作業の概要を示したものです。インストールしようとする環境にとってもっとも効率的なインストールを行うためにどの項目を決定する必要があるかを、この作業マップから判断してください。

表 2-1 作業マップ: Solaris ソフトウェアのインストールまたはアップグレード

作業	説明	参照先
初期インストールかアップグレードかを選択する	初期インストールまたはアップグレードのいずれかを選択します。	23 ページの「初期インストールかアップグレードか」
システム要件を検討する	インストールまたはアップグレードの最小要件をシステムが満たしているかどうかを判断します。	24 ページの「システム要件」
インストール方法を選択する	Solaris オペレーティング環境では、インストールやアップグレードをいくつかの方法で行うことができます。インストール環境にもっとも適した方法を選択してください。	第 3 章
ディスク容量の計画を立てて割り当てる	インストールする Solaris オペレーティング環境のコンポーネントに必要なディスク容量をシステムに割り当てます。	第 5 章
インストール媒体を選択する	Solaris ソフトウェアのインストールは、ローカル媒体から行うこともネットワークから行うこともできます。インストール環境にもっとも適したインストール媒体を選択してください。	25 ページの「ネットワークからインストールするか DVD または CD を使ってインストールするか」
システム情報を収集する	チェックリストやワークシートを使って、インストールやアップグレードに必要なすべての情報を収集します。	第 6 章
(省略可能) システム構成情報を事前設定する	インストールやアップグレードの際に情報を入力する手間を省くために、システム構成情報を事前に設定しておくことができます。	第 7 章
(省略可能) Solaris ソフトウェアをネットワークからインストールする準備を行う	Solaris ソフトウェアをネットワークからインストールする場合は、インストールサーバーを作成し、必要に応じてブートサーバーを作成し、ネットワークからインストールできるようにシステムを設定します。	第 12 章
(アップグレードのみ) アップグレード前に必要な作業を行う	システムのバックアップをとり、アップグレード時にディスク容量の再配置が行われるかどうかを判断し、Solaris Update リリースによって上書きされる可能性があるパッチを検索します。	第 8 章
インストールまたはアップグレードを行う	選択した Solaris インストール方法を使って Solaris ソフトウェアのインストールまたはアップグレードを行います。	インストールの詳細な手順を説明している章

初期インストールかアップグレードか

まず、初期インストールを行うのかアップグレードを行うのかを選択する必要があります。アップグレードを選択する場合は、Solaris オペレーティング環境がすでにシステム上で動作している必要があります。

初期インストール

初期インストールでは、システムのディスクが Solaris オペレーティング環境の新しいバージョンで上書きされます。システム上で Solaris オペレーティング環境がまだ動作していない場合は、初期インストールを行う必要があります。

システム上で Solaris オペレーティング環境がすでに動作している場合でも、初期インストールを行うことができます。ローカルに行なった変更を維持したい場合は、インストールを行う前にローカル変更のバックアップをとり、インストールが終わった後でローカル変更を復元する必要があります。

初期インストールは、Solaris のどのインストール方法を使っても実行できます。Solaris のインストール方法については、第 3 章を参照してください。

アップグレード

アップグレードでは、システムのディスクにある既存のファイルと Solaris オペレーティング環境の新しいバージョンがマージされます。アップグレードでは、既存の Solaris オペレーティング環境に対して行なった変更はできるだけ保存されます。

アップグレードできるシステムは、Solaris 2.6、Solaris 7、または Solaris 8 ソフトウェアが動作しているものです。システムで動作している Solaris のバージョンを確認するには、次のコマンドを入力します。

```
$ uname -a
```

どのインストール方法を使用しても、Solaris オペレーティング環境はアップグレードできます。Solaris Live Upgrade によるアップグレードの制限については、247 ページの「Solaris Live Upgrade のシステム要件」を参照してください。

注 - ディスクレスクライアントをアップグレードする場合は、`smosservice patch` コマンドを使用する必要があります。詳細な指示については、『Solaris のシステム管理 (基本編)』または `smosservice(1M)` のマニュアルページを参照してください。

システムのソフトウェアグループを、アップグレード時に別のソフトウェアグループに変更することはできません。たとえば、システムに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がインストールされている場合には、開発者システムサポートソフトウェアグループにアップグレードするオプションはありません。ただし、アップグレード中に、インストール済みのソフトウェアグループに属していないソフトウェアをシステムに追加することはできます。

すでに Solaris 9 オペレーティング環境を実行していて、システムに個別のパッチがインストールされている場合、Solaris 9 Update リリースへのアップグレードを行うと、パッチは次のようになります。

- Solaris 9 Update リリースの一部として組み込まれているパッチは自動的に再適用されます。これらのパッチをバックアウトすることはできません。
- システムにすでにインストールされているパッチのうち、Solaris 9 Update リリースに組み込まれていないパッチは削除されます。

パッチアナライザを使用すれば、Solaris 9 Update リリースにアップグレードしたときに、これらのパッチのうちどれが削除されるかを確認することができます。パッチアナライザの詳しい使い方については、369 ページの「Solaris Update リリースへのアップグレード」を参照してください。

システム要件

Solaris オペレーティング環境のインストールやアップグレードに必要な要件をシステムが満たしているかどうかを確認してください。

メモリー要件

Solaris オペレーティング環境をインストールまたはアップグレードするには、128M バイト以上のメモリーを用意することをお勧めします (最小では 96M バイト)。

注 - オプションのインストール機能の中には、メモリーが十分でないとは有効にできないものもあります。たとえば、メモリーが 96M バイトのシステムに DVD からインストールする場合、Solaris Web Start インストールプログラムのグラフィカルユーザーインタフェースではなく、Solaris Web Start のコマンド行インタフェースを使用する必要があります。

Solaris 9 INSTALLATION CD を使用するための要件

Solaris 9 INSTALLATION CD と Solaris Web Start プログラムを使ってアップグレードを行うためには、ファイルが保存されていないスライスがディスク上になければなりません。望ましいスライスは swap スライスですが、`/etc/vfstab` にリストされているアップグレード対象のルートスライスのどれにも属していないスライスであれば、どのスライスでもかまいません。このスライスのサイズは少なくとも 512M バイト必要です。

ネットワークからインストールするか DVD または CD を使ってインストールするか

DVD-ROM または CD-ROM ドライブにアクセスできるシステムへのインストールまたはアップグレードができるように、Solaris ソフトウェアは DVD または CD メディアで配布されます。

システムにローカルの DVD-ROM ドライブまたは CD-ROM ドライブがない場合や、Solaris ソフトウェアを複数のシステムにインストールする際に、それぞれのシステムに対してローカルドライブにディスクを挿入したくない場合は、リモートの DVD イメージまたは CD イメージからインストールするようにシステムを設定できます。

ネットワークからインストールする場合は、どの Solaris インストール方法でも使用できます。ただし、フラッシュインストール機能やカスタム JumpStart インストールを使ってネットワークからインストールを行うと、大規模の企業におけるインストールプロセスの一元化と自動化が可能になります。各インストール方法の詳細は、第 3 章を参照してください。

ネットワークから Solaris ソフトウェアをインストールする場合は、初期設定が必要です。ネットワークからインストールする場合の準備については、第 12 章を参照してください。

DVD メディアの使用

DVD メディアを使用している場合、ok プロンプトでブートするかどうかをたずねられたときには、常に次のコマンドを入力してください。

```
ok boot cdrom
```

第 3 章

Solaris インストール方法の選択

この章では、Solaris をインストールする方法について説明します。Solaris オペレーティング環境では、インストールやアップグレードをいくつかの方法で行うことができます。それぞれのインストール方法には、特定のインストール要件やインストール環境を意図したさまざまな機能があります。したがって、インストール環境にもっとも適した方法を選択してください。

- 27 ページの「Solaris Web Start」
- 28 ページの「Solaris suninstall プログラム」
- 28 ページの「カスタム JumpStart」
- 29 ページの「フラッシュインストール機能」
- 30 ページの「Solaris Live Upgrade の方法」
- 30 ページの「ファクトリ JumpStart」

Solaris Web Start

Solaris 9 DVD または Solaris 9 INSTALLATION CD に含まれる Solaris Web Start インストールプログラムは、グラフィカルユーザーインターフェース (GUI) でも、コマンド行インターフェース (CLI) でも起動できます。Solaris Web Start では、Solaris ソフトウェアや追加ソフトウェアのインストールやアップグレードに必要な手順がステップごとに示されます。システム管理者は、デフォルトオプションを使ってインストールすることも、カスタマイズオプションを使って必要なソフトウェアだけをインストールすることもできます。

Solaris Web Start では、Solaris オペレーティング環境や UNIX の初心者であっても、インストールの際に先に進んだり前に戻って必要な変更を簡単に行うことができます。インストール作業は複数の画面に分かれており、各画面にはユーザーが選択できるようにデフォルトの値が表示されます。

Solaris Web Start プログラムでは情報を入力する必要があるため、ユーザーはインストールプログラムと対話方式で処理を進める必要があります。したがって、システムによってはインストールやアップグレードを行う際には、このインストール方法が最適であるとは限りません。多数のシステムを対象としたバッチインストールには、カスタム JumpStart かフラッシュインストール機能を使用してください。

詳細は、第 14 章を参照してください。

Solaris suninstall プログラム

Solaris 9 SOFTWARE 1 of 2 CD に含まれる Solaris suninstall プログラムは、コマンド行インタフェース (CLI) で動作します。Solaris suninstall プログラムでは、Solaris 9 ソフトウェアのインストールやアップグレードに必要な手順がステップごとに示されます。このインストール方法は、96M バイト以上のメモリーがあり、国際ロケールを使用している場合に便利です。

Solaris suninstall プログラムは、Solaris オペレーティング環境ソフトウェアをインストールするだけです。このプログラムは、サードパーティアップグレードやネットワーク上でダウンロードできるソフトウェアなどを認識できません。したがって、Solaris オペレーティング環境をインストールした後にサードパーティアプリケーションをインストールする必要があります。また、インストール時にはシステム構成情報を入力するように求められます。このため、複数のシステムにインストールする場合は、suninstall は最適とは言えません。サードパーティアプリケーションのインストールには Solaris Web Start プログラムを使用できます。多数のシステムを対象としたバッチインストールには、カスタム JumpStart かフラッシュインストール機能を使用してください。

詳細は、第 15 章を参照してください。

カスタム JumpStart

カスタム JumpStart では、あらかじめ作成したプロファイルを使って、複数のシステムのインストールやアップグレードを自動的にかつ同時に行うことができます。プロファイルには、どのようにソフトウェアをインストールするかを定義します。さらに、インストール前とインストール後に実行する作業を、シェルスクリプトを使用して指定することができます。システムのインストールまたはアップグレードにどのプロファイルとスクリプトを使用するかを選択できます。カスタム JumpStart は、指定されたプロファイルとスクリプトに従ってシステムのインストールやアップグレードを行います。

Solaris オペレーティング環境とシェルに関する知識を持っていて、複数のシステムをインストールする必要がある場合には、カスタム JumpStart が最適であるかもしれません。

インストールするシステムが 2、3 台だけの場合には、このインストール方法が最適であるとは限りません。カスタム JumpStart インストールの準備に時間がかかるため、このインストール方法が時間の節約にならない可能性があるからです。

詳細は、第 23 章を参照してください。

フラッシュインストール機能

フラッシュインストール機能では、マスターシステムにインストールする構成を使用して、多数のシステムにインストールすることができます。それには、マスターシステムのインストールと構成を行なったあとに、マスターシステムからフラッシュアーカイブを作成する必要があります。フラッシュアーカイブは、必要に応じていくつでも作成できます。それぞれのシステムにインストールする際に、使用するフラッシュアーカイブを選択します。このインストール方法では、同じソフトウェアと構成を持つ多数のシステムを効率的にインストールできます。

フラッシュアーカイブを使用しない Solaris インストール方法では、各 Solaris パッケージが個別にインストールされます。パッケージベースのインストールではパッケージごとにパッケージマップの更新が必要になるため、時間がかかります。フラッシュアーカイブによるインストールは、個々の Solaris パッケージをインストールする場合よりもずっと早く終わります。

フラッシュアーカイブを使用するインストールは、どの Solaris インストール方法でも指定できます。Solaris Web Start と Solaris suninstall プログラムでは、インストールするフラッシュアーカイブを選択するように指示されます。カスタム JumpStart では、インストールするフラッシュアーカイブをプロファイルに指定します。Solaris Live Upgrade では、非アクティブブート環境にインストールするフラッシュアーカイブを指定します。

複数のシステムに多くの異なる構成でインストールしたい場合には、システムごとにフラッシュアーカイブが必要になります。フラッシュアーカイブはファイルサイズが大きいため、大量のディスク容量が必要です。さらに、フラッシュアーカイブをいったん作成したら、アーカイブを変更することはできません。多数の異なるインストール構成が必要であったり、インストール構成を変更する柔軟性を残しておきたい場合には、カスタム JumpStart インストールを使用することを検討してください。

詳細は、第 17 章を参照してください。

Solaris Live Upgrade の方法

Solaris Live Upgrade では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が行え、稼動中の環境のダウンタイムをなくすことができます。Solaris Live Upgrade は、GUI または CLI のどちらでも行えます。まず、複製ブート環境を作成する必要があります。この作業が終わったところで、アクティブでないブート環境 (非アクティブブート環境) でのフラッシュアーカイブをアップグレードするか、あるいは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できます。準備が整った時点で、この非アクティブブート環境をアクティブにします。次のリブート時には、この非アクティブブート環境がアクティブブート環境に切り替わります。何か問題が発生する場合は、本来のブート環境をアクティブにしてリブートするだけで元どおりに復元できます。

詳しい説明と操作方法については、第 29 章を参照してください。

ファクトリ JumpStart

ファクトリ JumpStart インストールでは、Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD をドライブに挿入して、システムの電源を入れるだけで、新しい SPARC システムに Solaris ソフトウェアを自動的にインストールできます。その際、システムの機種とディスクサイズに基づいてデフォルトのプロファイルが選択されます。システムにどのソフトウェアコンポーネントをインストールするかは、このプロファイルで決まります。システム構成情報を入力するように求められることはなく、インストールするソフトウェアを選択することはできません。

SPARC ベースの新しいシステムには、このインストール方法を使用する場合に欠かせない JumpStart ブートイメージがあらかじめインストールされています。これより古い SPARC ベースシステムの場合は、`re-preinstall(1M)` コマンドを使用すれば、JumpStart ブートイメージをシステムにインストールできます。

第 4 章

Solaris のインストールおよびアップグレードの準備

以下の章では、Solaris オペレーティング環境のインストールおよびアップグレードを行うための準備について説明します。

第 5 章	Solaris オペレーティング環境のインストールやアップグレードの際に必要なディスク容量を計画するための指針を示します。
第 6 章	Solaris のインストールを始める前にどのようなシステム情報を取得すべきかを説明します。
第 7 章	Solaris のインストールに必要なシステム構成情報を <code>sysidcfg</code> ファイルを使って事前設定する方法を説明します。
第 8 章	Solaris オペレーティング環境をアップグレードする前に必要な作業について説明します。

第 5 章

ディスク容量の割り当てに関する指針

この章では、Solaris オペレーティング環境のインストールやアップグレードに伴うディスク容量の計画に関して一般的な指針を説明します。この章の内容は次のとおりです。

- 33 ページの「ディスク容量に関する一般的な計画と推奨事項」
- 34 ページの「ソフトウェアグループごとの推奨ディスク容量」

ディスク容量に関する一般的な計画と推奨事項

Solaris 9 ソフトウェアをインストールする前にディスク容量の計画をたてて、システムに十分なディスク容量があるかどうかを判断することができます。必要なディスク容量が事前にわかれば、Solaris 9 ソフトウェアを実際にインストールする前にディスクを増設しておくことができます。アップグレードを行う場合は、61 ページの「ディスク容量の再配置を使用するアップグレード」を参照してください。

ディスク容量の計画のたて方は、ユーザーによって異なります。ディスク容量の計画にあたっては、次の一般的な事項を考慮に入れてください。

- 中国語、日本語、韓国語など、インストールする言語を追加するごとにディスク容量を増やします。
- 印刷やメールをサポートする場合は、`/var` ファイルシステムのディスク容量を増やします。
- クラッシュダンプ機能 `savecore (1M)` を使用する場合は、`/var` ファイルシステムの容量を物理メモリ一倍増します。
- サーバーがほかのシステム上のユーザーに対してホームディレクトリファイルシステムを提供する場合は、サーバーのディスク容量を増やします。ホームディレクトリは、通常デフォルトで `/export` ファイルシステムにあります。

- スワップ領域に 512M バイト以上を割り当てます。
- インストールする Solaris ソフトウェアグループ用にディスク容量を割り当てます。ソフトウェアグループごとの推奨ディスク容量は表 5-1 のとおりです。ディスク容量を計画する際には、選択したソフトウェアグループから個々のソフトウェアパッケージを個別に追加したり削除したりできることを覚えておいてください。
- 作成するファイルシステムの数を最小限に抑えます。デフォルトでは、Solaris インストール方法によってルート(/)とスワップ領域(/swap)だけが作成されます。OS サービスのためにディスク容量が割り当てられたときは、/export ディレクトリも作成されます。ファイルシステムを割り当てる場合には、将来の Solaris バージョンにアップグレードするときのために、現在必要な容量よりも 30% 多く割り当ててください。新しい Solaris リリースにアップグレードするたびに、直前のリリースに比べておおよそ 10% のディスク容量が追加で必要になります。ファイルシステムごとに 30% のディスク容量を追加しておくことによって、システムのスライスを切り直さなくても Solaris をアップグレードできます。
- 追加ソフトウェアやサードパーティソフトウェア用にディスク容量を増やします。

Solaris Live Upgrade を使用する場合は、248 ページの「Solaris Live Upgrade のディスク容量の要件」を参照してください。

ソフトウェアグループごとの推奨ディスク容量

Solaris ソフトウェアグループは Solaris パッケージの集まりです。それぞれのソフトウェアグループには、異なる機能やハードウェアドライバのサポートが含まれています。したがって、システムでどの機能を実行するかを考慮して、インストールするソフトウェアグループを選択する必要があります。

- 「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 – ネットワークに接続された Solaris システムと共通デスクトップ環境 (CDE) の起動と実行に必要な最小限のコードが含まれています。
- 「開発者システムサポート」ソフトウェアグループ – 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に、ソフトウェア開発のサポートが追加されたものです。ソフトウェア開発のサポートとして、ライブラリ、インクルードファイル、マニュアルページ、プログラミングツールなどが追加されています。ただし、コンパイラは含まれていません。
- 「全体ディストリビューション」ソフトウェアグループ – 開発者システムサポートソフトウェアグループに、サーバーに必要なソフトウェアが追加されたものです。
- 「全体ディストリビューションと OEM サポート」ソフトウェアグループ – 全体ディストリビューションソフトウェアグループに、ハードウェアドライバが追加されたものです。これには、インストール時にシステムに存在していないハードウェアのドライバも含まれます。

Solaris ソフトウェアのインストール時には、選択した Solaris ソフトウェアグループに対してパッケージを追加したり、削除したりすることができます。追加や削除するパッケージを選択する際には、ソフトウェアの依存関係や Solaris ソフトウェアがどのようにパッケージ化されているかを知っている必要があります。

次の表に、各 Solaris ソフトウェアグループごとに、推奨するディスク容量を示します。

注 – スワップ領域は、推奨ディスク容量に含まれています。

表 5-1 ソフトウェアグループごとの推奨ディスク容量

ソフトウェアグループ	推奨ディスク容量
全体ディストリビューションと OEM サポート	2.9G バイト
全体ディストリビューション	2.7G バイト
開発者システムサポート	2.4G バイト
エンドユーザーシステムサポート	2.0G バイト

第 6 章

インストールやアップグレードの前に 収集すべき情報

この章では、システムのインストールやアップグレードを行う前に収集すべきすべての情報が列挙されたチェックリストとワークシートについて説明します。

- 37 ページの「インストール用のチェックリスト」
- 38 ページの「インストール用のワークシート」
- 42 ページの「アップグレード用のチェックリスト」
- 43 ページの「アップグレード用のワークシート」

インストール用のチェックリスト

Solaris オペレーティング環境をインストールするための準備では、次のチェックリストを使用します。

- tip (1) 接続を使用して Solaris ソフトウェアをシステムにインストールする場合は、ウィンドウ表示が少なくとも横 80 桁、縦 24 行あることを確認します。
tip ウィンドウの現在の大きさを調べるには、`stty` コマンドを使用します。詳細は、`stty(1)` のマニュアルページを参照してください。
- システムがネットワークに接続されている場合は、Ethernet コネクタまたはそれに類似したネットワークアダプタがシステムに装着されていることを確認します。
- Solaris 9 INSTALLATION CD に入っている Solaris Web Start プログラムを使用してインストールする場合は、25 ページの「Solaris 9 INSTALLATION CD を使用するための要件」を参照してください。
- ディスク容量が十分あることを確認します。詳細は、第 5 章を参照してください。
- DVD メディアを使用している場合、`ok` プロンプトからのブートを求められたときには、常に次のコマンドを入力してください。**`boot cdrom`**
- 『Solaris 9 ご使用にあたって (SPARC 版)』やベンダーのリリースノートを参照して、使用するソフトウェアが新しい Solaris リリースでもサポートされていることを確認します。

- 『Solaris 9 Sun ハードウェアマニュアル』を参照して、使用するハードウェアがサポートされていることを確認します。
- システムに添付されている資料を参照して、使用するシステムやデバイスが Solaris リリースでサポートされていることを確認します。

インストール用のワークシート

Solaris オペレーティング環境のインストールに必要な情報を収集する際に、次のワークシートを使用します。ただし、ワークシートに記載されているすべての情報を収集する必要はありません。使用するシステムに関連する情報だけを収集してください。

注 - ネットワークに接続されないスタンドアロンシステムをインストールする場合は、* の部分を無視してください。

表 6-1 インストール用ワークシート

アップグレードに必要な情報	説明/例	答を記入する
ネットワーク	このシステムはネットワークに接続されていますか。	はい/いいえ
DHCP*	このシステムでは、DHCP (Dynamic Host Configuration Protocol) を使ってネットワークインタフェースを構成しますか。	はい/いいえ
ホスト名	選択したこのシステムのホスト名	
IP アドレス*	DHCP を使用しない場合は、このシステムの IP アドレスを指定します。 例 : 129.200.9.1	
サブネット*	DHCP を使用しない場合、このシステムはサブネットの一部ですか。 「はい」の場合は、サブネットのネットマスクを指定します。 例 : 255.255.0.0	はい/いいえ
IPv6*	このマシンで IPv6 を使用可能にしますか。	はい/いいえ

表 6-1 インストール用ワークシート (続き)

アップグレードに必要な情報	説明/例	答を記入する
Kerberos*	<p>このマシンに Kerberos セキュリティを構成しますか。</p> <p>「はい」の場合は、次の情報を収集します。</p> <p style="text-align: right;">デフォルトのレルム: 管理サーバー: 一次 KDC: (省略可能) 追加 KDC:</p>	はい/いいえ
ネームサービス*	このシステムではどのネームサービスを使用しますか。	NIS+/NIS/DNS/LDAP/使用しない
ドメイン名*	このシステムでネームサービスを使用する場合は、システムが属するドメインの名前を指定します。	
NIS+ および NIS*	<p>ネームサーバーを指定しますか、それともインストールプログラムにネームサーバーの検索を任せますか。</p> <p>ネームサーバーを指定する場合は、次の情報を指定します。</p> <p style="text-align: right;">サーバーのホスト名: サーバーの IP アドレス:</p>	指定/検索:
DNS*	<p>DNS サーバーの IP アドレスを指定します。DNS サーバーの IP アドレスを少なくとも 1 つ、最大 3 つまで指定します。</p> <p style="text-align: right;">サーバーの IP アドレス:</p> <p>DNS 検索を行うときに検索するドメインのリストを入力できます。</p> <p style="text-align: right;">検索ドメイン: 検索ドメイン: 検索ドメイン:</p>	
LDAP*	<p>LDAP プロファイルに関する次の情報を指定します。</p> <p style="text-align: right;">プロファイル名: プロファイルサーバーのホスト名: プロファイルサーバーの IP アドレス:</p>	

表 6-1 インストール用ワークシート (続き)

アップグレードに必要な情報	説明/例	答を記入する
デフォルトルート*	デフォルトルート (ゲートウェイ) を指定しますか、それとも Solaris Web Start にデフォルトルートの検索を任せますか。 デフォルトルートを指定する場合は、次の情報を指定します。 ルーターの IP アドレス:	指定/検索
時間帯	デフォルトの時間帯をどのように指定しますか。	地域 GMT との時間差 時間帯ファイル
ロケール	どの地域のサポートロケールをインストールしますか。	
電源管理システム	電源管理システムを使用しますか。	はい/いいえ
プロキシサーバーの構成* (Solaris Web Start の場合のみ)	インターネットに直接接続できますか、それともインターネットへのアクセスにプロキシサーバーを経由する必要がありますか。 プロキシサーバーを使用する場合は、次の情報を指定します。 ホスト: ポート:	直接接続/プロキシサーバー
自動的なリブートまたは CD/DVD 取り出し	ソフトウェアをインストールした後に自動的にリブートしますか。 ソフトウェアをインストールした後に CD/DVD を自動的に取り出しますか。	はい/いいえ はい/いいえ
ソフトウェアグループ	どの Solaris ソフトウェアグループをインストールしますか。	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 コアシステムサポート

表 6-1 インストール用ワークシート (続き)

アップグレードに必要な情報	説明/例	答を記入する
カスタムパッケージ選択	<p>インストールする Solaris ソフトウェアグループに対してソフトウェアパッケージの追加や削除を行いますか。</p> <p>注 - パッケージの追加や削除を行う場合には、ソフトウェアの依存関係や Solaris ソフトウェアがどのようにパッケージ化されているかを知っている必要があります。</p>	
64 ビット	64 ビットアプリケーションのサポートをインストールしますか。	はい/いいえ
ディスクの選択	<p>Solaris ソフトウェアをどのディスクにインストールしますか。</p> <p>例: c0t0d0</p>	
データの保存	Solaris ソフトウェアをインストールするために選択したディスク上の既存のデータを保存しますか。	はい/いいえ
ファイルシステムの自動配置	<p>ディスク上にあるファイルシステムを自動的に配置するために自動配置を使用しますか。</p> <p>「はい」の場合は、どのファイルシステムを自動的に配置しますか。</p> <p>例: /, /opt, /var</p> <p>「いいえ」の場合は、手作業でファイルシステムを配置する必要があります。</p>	はい/いいえ
リモートファイルシステムのマウント* (Solaris suninstall プログラムの場合のみ利用可能)	<p>このシステムからほかのファイルシステムにあるソフトウェアにアクセスする必要がありますか。</p> <p>必要な場合、リモートファイルシステムに関する次の情報を用意します。</p> <p>サーバー: プロファイルサーバーの IP アドレス: リモートファイルシステム: ローカルマウントポイント:</p>	はい/いいえ

アップグレード用のチェックリスト

Solaris オペレーティング環境をアップグレードするための準備では、次のチェックリストを使用します。

- tip (1) 接続を使用してシステムの Solaris ソフトウェアをアップグレードする場合は、ウィンドウ表示が少なくとも横 80 桁、縦 24 行あることを確認します。
tip ウィンドウの現在の大きさを調べるには、`stty` コマンドを使用します。詳細については、`stty (1)` のマニュアルページを参照してください。
- システムがネットワークに接続されている場合は、Ethernet コネクタまたはそれに類似したネットワークアダプタがシステムに装着されていることを確認します。
- Solaris 9 INSTALLATION CD に入っている Solaris Web Start プログラムを使用する場合は、ディスクに 512M バイトのスライスが存在することを確認します。
- Solaris Live Upgrade を使用する場合は、新しいブート環境を作成してアップグレードするためのリソース要件を決定します。詳細は、第 31 章を参照してください。
- Solaris Live Upgrade と Solaris ボリュームマネージャを使用する場合は、ルートのみラーリングをオフにします。詳細は、251 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」を参照してください。
- ディスク容量が十分あることを確認します。詳細は、第 5 章を参照してください。
- DVD メディアを使用している場合、`ok` プロンプトからブートするように求められたときには、常に次のコマンドを入力してください。**`boot cdrom`**
- 『Solaris 9 ご使用にあたって (SPARC 版)』やベンダーのリリースノートを参照して、使用するソフトウェアが新しい Solaris リリースでもサポートされていることを確認します。
- 『Solaris 9 Sun ハードウェアマニュアル』を参照して、使用するハードウェアがサポートされていることを確認します。
- システムに添付されている資料を参照して、使用するシステムやデバイスが Solaris リリースでサポートされていることを確認します。
- ベンダーやサードパーティのソフトウェア資料を参照して、アップグレードに関するその他の情報を入手します。
- 使用可能なパッチで必要なパッチがあるかを確認します。最新のパッチリストは <http://sunsolve.sun.com> から入手できます。
- Prestoserve ソフトウェアがシステムに存在するかを確認します。`init 0` コマンドを使ってシステムをシャットダウンしてからアップグレードプロセスを開始すると、データが失われる恐れがあります。シャットダウンについての説明は、Prestoserve の資料を参照してください。

アップグレード用のワークシート

Solaris オペレーティング環境のアップグレードに必要な情報を収集する際には、次のワークシートを使用します。ただし、ワークシートに記載されているすべての情報を収集する必要はありません。使用するシステムに関連する情報だけを収集してください。アップグレードをネットワークを使用して行う場合は、インストールプログラムが現在のシステム構成から情報を取得します。

ホスト名や IP アドレスのような、システムの基本的な識別情報は変更できません。インストールプログラムによってシステムの基本的な識別情報を入力するように求められる場合がありますが、元の値を入力する必要があります。Solaris Web Start プログラムを使用してアップグレードする場合は、そのような識別情報の値をどれか変更しようとするするとアップグレードは失敗します。

注 - ネットワークに接続されないスタンドアロンシステムをインストールする場合は、* の部分を無視してください。

表 6-2 アップグレード用ワークシート

アップグレードに必要な情報	説明/例	答を記入する
ネットワーク	このシステムはネットワークに接続されていますか。	はい/いいえ
DHCP*	このシステムでは、DHCP (Dynamic Host Configuration Protocol) を使ってネットワークインタフェースを構成しますか。	はい/いいえ
ホスト名	選択したこのシステムのホスト名	
IP アドレス*	DHCP を使用しない場合は、このシステムの IP アドレスを指定します。 例 : 129.200.9.1	
サブネット*	DHCP を使用しない場合、このシステムはサブネットの一部ですか。 「はい」の場合は、サブネットのネットマスクを指定します。 例 : 255.255.0.0	はい/いいえ
IPv6*	このマシンで IPv6 を使用可能にしますか。	はい/いいえ

表 6-2 アップグレード用ワークシート (続き)

アップグレードに必要な情報	説明/例	答を記入する
Kerberos*	<p>このマシンに Kerberos セキュリティを構成しますか。</p> <p>「はい」の場合は、次の情報を収集します。</p> <p style="text-align: right;">デフォルトのレルム: 管理サーバー: 一次 KDC: (省略可能) 追加 KDC:</p>	はい/いいえ
ネームサービス*	このシステムではどのネームサービスを使用しますか。	NIS+/NIS/DNS/LDAP/使用しない
ドメイン名*	このシステムでネームサービスを使用する場合は、システムが属するドメインの名前を指定します。	
NIS+ および NIS*	<p>ネームサーバーを指定しますか、それともインストールプログラムにネームサーバーの検索を任せますか。</p> <p>ネームサーバーを指定する場合は、次の情報を指定します。</p> <p style="text-align: right;">サーバーのホスト名: サーバーの IP アドレス:</p>	指定/検索
DNS*	<p>DNS サーバーの IP アドレスを指定します。DNS サーバーの IP アドレスを少なくとも 1 つ、最大 3 つまで指定します。</p> <p style="text-align: right;">サーバーの IP アドレス:</p> <p>DNS 検索を行うときに検索するドメインのリストを入力できます。</p> <p style="text-align: right;">検索ドメイン:</p>	
LDAP*	<p>LDAP プロファイルに関する次の情報を指定します。</p> <p style="text-align: right;">プロファイル名: プロファイルサーバーのホスト名: プロファイルサーバーの IP アドレス:</p>	

表 6-2 アップグレード用ワークシート (続き)

アップグレードに必要な情報	説明/例	答を記入する
デフォルトルート*	デフォルトルート (ゲートウェイ) を指定しますか、それとも Solaris Web Start にデフォルトルートの検索を任せますか。 デフォルトルートを指定する場合は、次の情報を指定します。 ルーターの IP アドレス:	指定/検索:
時間帯	デフォルトの時間帯をどのように指定しますか。	地域 GMT との時間差 時間帯ファイル
ロケール	どの地域のサポートをインストールしますか。	
電源管理システム	電源管理システムを使用しますか。	はい/いいえ
プロキシサーバーの構成* (Solaris Web Start の場合のみ)	インターネットに直接接続できますか、それともインターネットへのアクセスにプロキシサーバーを経由する必要がありますか。 プロキシサーバーを使用する場合は、次の情報を指定します。 ホスト: ポート:	直接接続/プロキシサーバー
自動的なりポートまたは CD/DVD 取り出し	ソフトウェアをインストールした後に自動的にリポートしますか。 ソフトウェアをインストールした後に CD/DVD を自動的に取り出しますか。	はい/いいえ はい/いいえ
ディスク容量の再割り当て	ディスク上にあるファイルシステムを自動的に再配置するために自動配置を使用しますか。 「はい」の場合は、どのファイルシステムを自動配置しますか。 例: /, /opt, /var 「いいえ」の場合は、ファイルシステム構成用の情報を指定する必要があります。	はい/いいえ
64 ビット	64 ビットアプリケーションのサポートをインストールしますか。	はい/いいえ

第 7 章

システム構成情報の事前設定

この章では、システム構成情報の事前設定について説明します。事前に設定を行うと、Solaris オペレーティング環境をインストールする際に、システム構成情報の入力を求めるプロンプトが表示されません。この章では、さらに、電源管理システム情報の事前設定についても説明します。この章の内容は次のとおりです。

- 47 ページの「システム構成情報を事前設定することの利点」
- 48 ページの「システム構成情報の事前設定方法」
- 49 ページの「sysidcfg ファイルによる事前設定」
- 54 ページの「ネームサービスによる事前設定」
- 57 ページの「電源管理情報の事前設定」

システム構成情報を事前設定することの利点

どのインストール方法でも、周辺機器、ホスト名、IP (インターネットプロトコル) アドレス、ネームサービスなどのシステム構成情報が必要です。インストールプログラムは、構成情報の入力を求める前に、まず sysidcfg ファイルの情報を調べ、次にネームサービスデータベースの情報を調べます。

Solaris Web Start プログラム、Solaris suninstall プログラム、およびカスタム JumpStart インストールプログラムは、事前設定されたシステム構成情報を検出すると、その情報の入力を求めることはありません。たとえば、複数のシステムに Solaris 9 ソフトウェアをインストールする場合、時間帯を sysidcfg ファイル中またはネームサービスデータベース中に指定しておくと、Solaris 9 ソフトウェアのインストール時に時間帯を入力する手間が省けます。

システム構成情報の事前設定方法

システム構成情報を事前設定するには、2つの方法があります。システム構成情報は以下の場所に設定できます。

- `sysidcfg` ファイル (リモートシステム上またはフロッピーディスク)
- 自分のサイトで使用しているネームサービスデータベース

表 7-1 を使って、システム構成情報を事前設定するための方法を決定してください。

表 7-1 システム構成情報を事前設定するための方法

事前設定できるシステム構成情報	<code>sysidcfg</code> ファイルでの事前設定の可否	ネームサービスでの事前設定の可否
ネームサービス	可能	可能
ドメイン名	可能	不可
ネームサーバー	可能	不可
ネットワークインタフェース	可能	不可
ホスト名	可能 ¹	可能
IP アドレス	可能 ¹	可能
ネットマスク	可能	不可
DHCP	可能	不可
IPv6	可能	不可
デフォルトのルーター	可能	不可
root パスワード	可能	不可
セキュリティポリシー	可能	不可
インストールプログラムとデスクトップで表示する言語 (ロケール)	可能	可能 (NIS または NIS+ の場合) 不可 (DNS または LDAP の場合)
端末タイプ	可能	不可
時間帯	可能	可能
日付と時刻	可能	可能
電源管理システム (autoshtutdown) ²	不可	不可

1. この情報はシステムに固有なため、各システム用に異なる `sysidcfg` ファイルを作成するよりも、ネームサーバーを編集してください。

2. このシステム構成情報は、`sysidcfg` ファイルまたはネームサービスでは事前設定できません。詳細は、57 ページの「電源管理情報の事前設定」を参照してください。

sysidcfg ファイルによる事前設定

`sysidcfg` ファイルに一連のキーワードを指定すると、システムを事前設定できます。表 7-2 は、これらのキーワードを示しています。

異なる構成情報を必要とするシステムごとに、固有の `sysidcfg` ファイルを作成する必要があります。すべてのシステムに同じ時間帯を割り当てる場合は、同じ `sysidcfg` ファイルを使用して、一連のシステムに時間帯を事前設定することができます。ただし、これらの各システムに異なる `root` (スーパーユーザー) パスワードを事前設定する場合は、各システムに固有の `sysidcfg` ファイルを作成する必要があります。

`sysidcfg` ファイルは、次のどちらかに置くことができます。

- NFS ファイルシステム – `sysidcfg` ファイルを共有 NFS ファイルシステムに置く場合は、ネットワークからインストールできるようにシステムを設定するときに、`add_install_client(1M)` コマンドの `-p` オプションを指定する必要があります。Solaris 9 ソフトウェアのインストール時に `sysidcfg` ファイルが検索される場所を指定する必要があります。
- UFS フロッピーディスク – `sysidcfg` ファイルをフロッピーディスクのルートディレクトリ (/) に置きます。

注 – カスタム JumpStart インストールを行なっていて、フロッピーディスク上の `sysidcfg` ファイルを使用したい場合は、プロファイルフロッピーディスク上に `sysidcfg` ファイルを置く必要があります。プロファイルフロッピーディスクを作成するには、149 ページの「スタンドアロンシステム用のプロファイルフロッピーディスクの作成」を参照してください。

1つのディレクトリまたはフロッピーディスクには、1つの `sysidcfg` ファイルだけを入れることができます。複数の `sysidcfg` ファイルを作成する場合は、各ファイルを異なるディレクトリまたは異なるフロッピーディスクに置く必要があります。

sysidcfg ファイルの構文規則

sysidcfg ファイルで使用するキーワードには、非依存型と依存型の 2 種類があります。依存型キーワードは、非依存型キーワード内でのみ固有であることが保証されています。依存型キーワードは、対応する非依存型キーワードによって識別される場合にのみ存在します。

次の例では、name_service が非依存型キーワードであり、domain_name と name_server が依存型キーワードです。

```
name_service=NIS {domain_name=marquee.central.sun.com
name_server=connor(129.152.112.3)}
```

構文規則	例
非依存型キーワードは任意の順序で指定可能	<pre>pointer=MS-S display=ati {size=15-inch}</pre>
キーワードは大文字と小文字のどちらでもよい	<pre>TIMEZONE=US/Central</pre>
関連する非依存型キーワードを結合するには、すべての依存型キーワードを中括弧 {} で囲む	<pre>name_service=NIS {domain_name=marquee.central.sun.com name_server=connor(129.152.112.3)}</pre>
値は単一引用符 (') または二重引用符 (") で囲んで指定可能	<pre>network_interface='none'</pre>
キーワードは 1 回だけ指定可能。キーワードを複数回指定した場合は最初のキーワードだけが有効	<pre>network_interface=none network_interface=le0</pre>

sysidcfg ファイルキーワード

表 7-2 に、sysidcfg ファイルで使用できるキーワードを示します。

表 7-2 sysidcfg で使用できるキーワード

構成情報	キーワード	使用例または設定値
ネームサービス、ドメイン名、ネームサーバー	name_service=NIS, NIS+, DNS, LDAP, NONE	

表 7-2 sysidcfg で使用できるキーワード (続き)

構成情報	キーワード	使用例または設定値
	NIS と NIS+ 用オプション: {domain_name=domain_name name_server=hostname(ip_address)}	name_service=NIS {domain_name=west.arp.com name_server=timber(129.221.2.1)} name_service=NIS+ {domain_name=west.arp.com.name_server=timber(129.221.2.1)}
	DNS 用オプション: {domain_name=domain_name name_server=ip_address, ip_address, ip_address (最大 3 個) search=domain_name, domain_name, domain_name, domain_name, domain_name (最大 6 個、合計の長さが 250 文字以下)}	name_service=DNS {domain_name=west.arp.com name_server=10.0.1.10, 10.0.1.20 search=arp.com, east.arp.com} 注 - name_service は 1 つの値だけを選択します。必要に応じて、domain_name と name_server キーワードのどちらか一方または両方を設定するか、あるいはどちらも含めません。どちらのキーワードも使用しない場合、中括弧 {} は省略します。
	LDAP 用オプション: {domain_name=domain_name profile=profile_name profile_server= ip_address}	name_service=LDAP {domain_name=west.arp.com profile=default profile_server=129.221.2.1}
ネットワークインタフェース、ホスト名、IP アドレス、ネットマスク、DHCP、IPv6	network_interface=NONE, PRIMARY, or value 値の説明: ■ NONE は、ネットワーク接続をオフにします。 ■ PRIMARY は、システム上に存在するインタフェースの中で最初に稼動する非ループバックインタフェースです。順序は "ifconfig" の場合と同様に、稼動しているインタフェースが存在しない場合には、最初の非ループバックインタフェースが使用されることとなります。非ループバックインタフェースが見つからない場合は、システムは NON-NETWORKED に設定されます。 ■ value には、le0 や hme0 などのインタフェースを指定します。	network_interface=primary {dhcp protocol_ipv6=yes_or_no}
	DHCP を使用する場合、次のように指定する。 {dhcp protocol_ipv6= yes_or_no}	network_interface=primary {dhcp protocol_ipv6=yes}

表 7-2 sysidcfg で使用できるキーワード (続き)

構成情報	キーワード	使用例または設定値
	DHCP を使用しない場合、次のように指定する。{hostname= <i>host_name</i> default_route= <i>ip_address</i> ip _address= <i>ip_address</i> netmask= <i>netmask</i> protocol_ipv6= <i>yes_or_no</i> }	network_interface=le0 {hostname=feron default_route=129.146.88.1 ip_address =129.146.88.210 netmask=255.255.0.0 protocol_ipv6=no} 注 - network_interface は、1 つの値だけを選択します。必要に応じて、hostname、ip_address、netmask キーワードのどれかを組み合わせさせて設定するか、あるいはどれも設定しません。どのキーワードも使用しない場合、中括弧 {} は省略します。 注 - DHCP を使用しない場合、protocol_ipv6 と default_route は指定する必要がありません。ただし、JumpStart インストールの場合、この時点で protocol_ipv6 を指定する必要があります。指定がないと、後で対話方式で入力を要求されます。
root パスワード	root_password= <i>root_password</i>	/etc/shadow にある暗号化された文字列
セキュリティポリシー	security_policy=kerberos, NONE Kerberos 用オプション:{default_realm=FQDN admin_server=FQDN kdc=FQDN1, FQDN2, FQDN3} (FQDN は完全修飾ドメイン名です。)	security_policy=kerberos {default_realm=Yoursite.COM admin_server=krbadm.Yoursite.COM kdc=kdc1.Yoursite.COM, kdc2.Yoursite.COM} 注 - 最大 3 つの鍵発行センター (KDC) をリストできます。少なくとも 1 つは必須です。
インストールプログラムとデスクトップで表示する言語	system_locale= <i>locale</i>	有効なロケール値が、/usr/lib/locale ディレクトリまたは第 40 章にあります。
端末タイプ	terminal= <i>terminal_type</i>	有効な端末値が /usr/share/lib/terminfo ディレクトリのサブディレクトリにあります。
時間帯	timezone= <i>timezone</i>	有効な時間帯値が /usr/share/lib/zoneinfo ディレクトリのサブディレクトリとファイルにあります。時間帯値は /usr/share/lib/zoneinfo ディレクトリからの相対パス名です。たとえば、米国の山岳部標準時の時間値は US/Mountain です。日本の時間帯値は Japan です。また、有効な Olson 時間帯も指定できます。

表 7-2 sysidcfg で使用できるキーワード (続き)

構成情報	キーワード	使用例または設定値
日付と時刻	timeserver=localhost, <i>hostname</i> , <i>ip_addr</i>	localhost を指定した場合は、そのシステムの時刻が正しいものと仮定し、時刻が設定されます。ネームサービスを実行してなくて、あるシステムの <i>hostname</i> または <i>ip_addr</i> を指定した場合は、そのシステムの時刻を使って時刻が設定されます。

▼ sysidcfg 構成ファイルを作成する方法

1. テキストエディタを使って sysidcfg という名前のファイルを作成します。
2. 必要な sysidcfg のキーワードを入力します。
3. sysidcfg ファイルを保存します。

注 - 複数の sysidcfg ファイルを作成する場合は、それぞれのファイルを別々のディレクトリまたは別々のフロッピーディスクに保存する必要があります。

4. クライアントから次のファイルシステムまたはディレクトリを介して sysidcfg ファイルにアクセスできるようにします。
 - 共有 NFS ファイルシステム。ネットワークからインストールできるようにシステムを設定するには、add_install_client(1M) に -p オプションを指定します。
 - UFS フロッピーディスク上のルート (/) ディレクトリ。

sysidcfg ファイルの例

次に、sysidcfg ファイルの例を示します。これらのシステムのホスト名、IP アドレス、およびネットマスクは、ネームサービスを編集することにより、すでに事前設定されています。このファイルにはすべてのシステム構成情報が事前設定されているので、カスタム JumpStart プロファイルを使ってカスタム JumpStart インストールを実行できます。

```
system_locale=en_US
timezone=US/Central
terminal=sun-cmd
timeserver=localhost
name_service=NIS {domain_name=marquee.central.sun.com
                  name_server=connor(129.152.112.3) }
root_password=m4QPOWNY
```

ネームサービスによる事前設定

次の表は、システム構成情報を事前設定するために編集および入力を行う必要があるネームサービスデータベースの概要を示したものです。

事前設定するシステム情報	ネームサービスデータベース
ホスト名と IP アドレス	hosts
日付と時刻	hosts。インストール対象のシステムに日付と時刻を提供するシステムのホスト名で timehost というホストの別名を持つマシンを使用する。
時間帯	timezone
ネットマスク	netmasks

DNS や LDAP のネームサービスでは、システムのロケールを事前設定することはできません。NIS や NIS+ のネームサービスを使用する場合は、そのネームサービスの手順に従ってシステムのロケールを事前設定してください。

- 54 ページの「NIS を使ってロケールを事前設定する方法」
- 56 ページの「NIS+ を使ってロケールを事前設定する方法」

▼ NIS を使ってロケールを事前設定する方法

1. ネームサーバー上でスーパーユーザーになります。
2. /var/yp/Makefile ファイルを編集して、ローカルマップを追加します。
 - a. エントリの後に、以下を追加します。

```
locale.time: $(DIR)/locale
    -@if [ -f $(DIR)/locale ]; then \
        sed -e "/^#/d" -e s/#.*$$// $(DIR)/locale \
        | awk '{for (i = 2; i<=NF; i++) print $$i, $$0}' \
        | $(MAKEDBM) - $(YPDBDIR)/$(DOM)/locale.byname; \
        touch locale.time; \
        echo "updated locale"; \
        if [ ! $(NOPUSH) ]; then \
            $(YPPUSH) locale.byname; \
            echo "pushed locale"; \
        else \
            : ; \
        fi \
    else \
        echo "couldn't find $(DIR)/locale"; \
```

fi

- b. 文字列 `all:` を検索し、変数リストの最後に `locale` という語を挿入します。

```
all: passwd group hosts ethers networks rpc services protocols \  
    netgroup bootparams aliases publickey netid netmasks c2secure \  
    timezone auto.master auto.home locale
```

- c. ファイルの後方にある同じようなエントリの後に、文字列 `locale:` `locale.time` を追加します。

```
passwd: passwd.time  
group: group.time  
hosts: hosts.time  
ethers: ethers.time  
networks: networks.time  
rpc: rpc.time  
services: services.time  
protocols: protocols.time  
netgroup: netgroup.time  
bootparams: bootparams.time  
aliases: aliases.time  
publickey: publickey.time  
netid: netid.time  
passwd.adjunct: passwd.adjunct.time  
group.adjunct: group.adjunct.time  
netmasks: netmasks.time  
timezone: timezone.time  
auto.master: auto.master.time  
auto.home: auto.home.time  
locale: locale.time
```

- d. ファイルを保存します。

3. `/etc/locale` というファイルを作成し、ドメインまたは特定のシステムに対して1つのエントリを作成します。

```
locale domain_name
```

または

```
locale system_name
```

注 – 使用できるロケール値のリストについては、第 40 章を参照。

たとえば次の行は、`worknet.com` ドメインに対してデフォルト言語として日本語を指定しています。

```
ja worknet.com
```

たとえば次の行は、`charlie` というシステムに対してデフォルトロケールとして `ja_JP.UTF-8` ロケールを指定しています。

```
ja_JP.UTF-8
```

注 – ロケールは、Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD に入っています。

4. マップを作成します。

```
# cd /var/yp; make
```

これでドメインまたは locale マップで個別に指定したシステムは、デフォルトのロケールを使用するように設定されました。ここで指定したデフォルトのロケールは、インストール時に使用されるとともに、システムのリブート後のデスクトップでも使用されます。

▼ NIS+ を使ってロケールを事前設定する方法

この手順は、NIS+ ドメインが設定されていると仮定しています。NIS+ ドメインの設定については、『Solaris のシステム管理 (ネーミングとディレクトリサービス: DNS、NIS、LDAP 編)』を参照してください。

1. ネームサーバーに、スーパーユーザーまたは NIS+ admin グループのユーザーとしてログインします。
2. locale テーブルを作成します。

```
# nistbladm -D access=og=rmcd,nw=r -c locale_tbl name=SI,nogw=locale=,nogw= comment=,nogw= locale.org_dir.`nisdefaults -d`
```

3. locale に必要なエントリを追加します。

```
# nistbladm -a name=name locale=locale comment=comment locale.org_dir.`nisdefaults -d`
```

<i>name</i>	デフォルトのロケールを事前設定したいドメイン名または特定のシステム名。
<i>locale</i>	システムにインストールし、システムのリブート後にデスクトップ上で使用するロケール。使用できるロケール値のリストについては、第 40 章を参照。
<i>comment</i>	コメントフィールド。複数の単語を使ったコメントは、前後を二重引用符で囲むこと。

注 – ロケールは、Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD に入っています。

これでドメインまたは locale テーブルで個別に指定したシステムは、デフォルトロケールを使用するように設定されました。ここで指定したデフォルトロケールは、インストール時に使用されるとともに、システムのリポート後のデスクトップでも使用されます。

電源管理情報の事前設定

Solaris の電源管理ソフトウェアを使用すると、システムが 30 分間アイドル状態になると自動的にシステム状態を保存し電源を切ることができます。EPA の省電力 (Energy Star) ガイドラインのバージョン 2 に準拠したシステム (sun4u システムなど) に Solaris 9 ソフトウェアをインストールするときは、デフォルトで電源管理ソフトウェアもインストールされます。インストール後のリポート時に、この電源管理ソフトウェアを有効または無効にするかを尋ねられます。

対話式インストールを実行している場合は、電源管理情報を事前設定してプロンプトを回避する方法はありません。カスタム JumpStart インストールでは、`finish` スクリプトを使ってシステムに `/autoshtutdown` または `/noautoshtutdown` ファイルを作成することで、電源管理情報を事前設定できます。システムのリポート時に、`/autoshtutdown` は電源管理ソフトウェアを有効にし、`/noautoshtutdown` ファイルは電源管理ソフトウェアを無効にします。

たとえば、`finish` スクリプトに次の行を入れておくと電源管理ソフトウェアが有効になり、システムリポート後にプロンプトが表示されないようにすることができます。

```
touch /a/autoshtutdown
```

`finish` スクリプトの詳細は、167 ページの「`finish` スクリプトの作成」を参照してください。

第 8 章

Solaris オペレーティング環境のアップグレード

この章では、Solaris オペレーティング環境をアップグレードする前に行う必要がある作業についての情報や手順を説明します。

- 59 ページの「アップグレード」
- 60 ページの「Solaris Live Upgrade の利用」
- 60 ページの「カスタム JumpStart によるアップグレード」
- 61 ページの「ディスク容量の再配置を使用するアップグレード」
- 62 ページの「アップグレード前のシステムのバックアップ」

アップグレード

アップグレードでは、システムのディスクにある既存のファイルと Solaris オペレーティング環境の新しいバージョンがマージされます。アップグレードでは、既存の Solaris オペレーティング環境に対して行なった変更はできるだけ保存されます。

アップグレードできるシステムは、Solaris 2.6、Solaris 7、または Solaris 8 ソフトウェアが動作しているものです。システムで動作している Solaris のバージョンを確認するには、次のコマンドを入力します。

```
$ uname -a
```

Solaris オペレーティング環境のアップグレードは、任意のインストール方法で行えます。Solaris Live Upgrade によるアップグレードの制限については、247 ページの「Solaris Live Upgrade のシステム要件」を参照してください。

注 - ディスクレスクライアントをアップグレードする場合は、smosservice patch コマンドを使用する必要があります。詳細は、『Solaris のシステム管理 (基本編)』または smosservice(1M) のマニュアルページを参照してください。

システムのソフトウェアグループを、アップグレード時に別のソフトウェアグループに変更することはできません。たとえば、システムに 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がインストールされている場合には、開発者システムサポートソフトウェアグループにアップグレードするオプションはありません。ただし、アップグレード中に、インストール済みのソフトウェアグループに属していないソフトウェアをシステムに追加することはできません。

すでに Solaris 9 オペレーティング環境を実行していて、システムに個別のパッチがインストールされている場合、Solaris 9 Update リリースへのアップグレードを行うと、パッチは次のようになります。

- Solaris 9 Update リリースの一部として組み込まれているパッチは自動的に再適用されます。これらのパッチをバックアウトすることはできません。
- システムにすでにインストールされているパッチのうち、Solaris 9 Update リリースに組み込まれていないパッチは削除されます。

パッチアナライザを使用すれば、Solaris 9 Update リリースにアップグレードしたときに、これらのパッチのうちどれが削除されるかを確認することができます。パッチアナライザの詳しい使い方については、369 ページの「Solaris Update リリースへのアップグレード」を参照してください。

Solaris Live Upgrade の利用

Solaris Live Upgrade を利用すると、複製された非アクティブオペレーティング環境をアップグレードすることによりオペレーティングシステムのダウンタイムを減らすことができます。

Solaris Live Upgrade の計画と実施の方法については、第 29 章を参照してください。

カスタム JumpStart によるアップグレード

カスタム JumpStart インストールを使用してアップグレードを行う場合は、カスタム JumpStart プロファイルに `install_type upgrade` を指定する必要があります。

さらに、実際にアップグレードを行う前に、システムの現在のディスク構成およびシステムに現在インストールされているソフトウェアに対して、カスタム JumpStart プロファイルが目的どおりのことを実行しようとしているかを確認する必要があります。プロファイルをテストするには、アップグレードするシステムで `pinstall -D`

コマンドを使用します。ディスク構成ファイルを使用してアップグレード用プロファイル进行测试することはできません。アップグレード用プロファイルのテストについての詳細は、159 ページの「プロファイルのテスト」を参照してください。

ディスク容量の再配置を使用するアップグレード

Solaris Web Start のアップグレードオプション、Solaris `suninstall` プログラムおよびカスタム JumpStart プログラムにはディスクの容量を再配置する機能があります。アップグレードするのに十分な容量が現在のファイルシステムにない場合、ディスク容量を割り当て直すことができます。アップグレードに伴ってファイルシステムの容量を増やす必要があるのは、次のような場合です。

- 新しいリリースで、システムに現在インストールされている Solaris ソフトウェアグループに新たにソフトウェアが追加されている。特定のソフトウェアグループに含まれる新しいソフトウェアは、インストールの対象となるようにアップグレード時に自動的に選択される。
- 新しいリリースに、システム上の既存のソフトウェアよりもサイズが大きいソフトウェアが存在する。

自動再配置機能を使用すると、ファイルシステムに必要な容量を確保するようにディスク容量の再配置が行われます。自動再配置機能では、デフォルトの制約に基づいて容量の再配置が試みられます。このため、この機能によって容量の再配置が行われない場合は、ファイルシステムの制約を変更する必要があります。

注 – 自動再配置機能には、ファイルシステムの容量を増やす能力はありません。自動再配置機能による容量の再配置では、変更の必要なファイルシステムのバックアップ、ファイルシステムの変更に基づくディスクパーティションの再分割、およびファイルシステムの復元がアップグレードの前に行われます。

- Solaris Web Start の自動配置機能がディスク容量をどのように再配置するかを決定できない場合は、Solaris `suninstall` プログラムまたはカスタム JumpStart プログラムを使用してアップグレードを行う必要があります。
- Solaris 9 `suninstall` プログラムの自動配置機能が、ディスク容量をどのように再配置するかを決定できない場合は、移動または変更してもよいファイルシステムを指定してから再び自動配置機能を実行します。
- カスタム JumpStart によるアップグレードでアップグレード用プロファイルを作成する際に、ディスク容量が問題になることがあります。現在のファイルシステムに十分なディスク容量がない場合は、`backup_media` と `layout_constraint` キーワードを使用してディスク容量を再配置することができます。`backup_media` と `layout_constraint` キーワードの使用例については、例 23-4 を参

照してください。

アップグレード前のシステムのバックアップ

Solaris オペレーティング環境のアップグレードを行う前に既存システムのバックアップを行う必要があります。ファイルシステムをテープなどの取り外し可能媒体にコピーすれば、データの損失や損傷、破壊などを防止できます。システムのバックアップ手順についての詳細は、『Solaris のシステム管理 (基本編)』を参照してください。

第 9 章

ネットワークインストールの準備

以下の章では、DVD メディアや CD メディアを利用するのではなくネットワークを使用して Solaris ソフトウェアをインストールする場合のシステム設定の方法を説明します。

第 10 章	インストールサーバーから Solaris ソフトウェアをインストールする方法の概要と計画について説明します。
第 11 章	DVD メディアからインストールサーバーに Solaris ソフトウェアをコピーする手順を説明します。
第 12 章	CD メディアからインストールサーバーに Solaris ソフトウェアをコピーする手順を説明します。
第 13 章	ネットワークインストールを設定するコマンドについて説明します。

第 10 章

ネットワークインストールの準備 (概要)

この章では、DVD または CD メディアからではなくネットワークから Solaris ソフトウェアをインストールする場合にネットワークとシステムをどのように設定する必要があるかを説明します。

ネットワークインストールの計画

この節では、ネットワークからインストールを行う前に認識しておくべき事柄を説明します。ネットワークインストールでは、Solaris 9 ディスクイメージにアクセスできる「インストールサーバー」というシステムから Solaris ソフトウェアをインストールできます。その場合には、まず Solaris 9 DVD または CD メディアの内容をインストールサーバーのハードディスクにコピーする必要があります。これで、任意の Solaris インストール方法を使用して Solaris ソフトウェアをネットワーク上でインストールできます。

ネットワークインストールに必要なサーバー

ネットワーク上で Solaris オペレーティング環境をインストールするには、次のサーバーがネットワーク上に存在する必要があります。

- インストールサーバー – Solaris 9 ディスクイメージ (CD イメージまたは DVD イメージ) が格納された、ネットワークに接続されたシステム。このシステムからネットワーク上の別のシステムに Solaris 9 をインストールできます。インストールサーバーを作成するには、以下のメディアの 1 つからイメージをコピーします。
 - Solaris 9 DVD
 - Solaris 9 SOFTWARE 1 of 2 CD および Solaris 9 SOFTWARE 2 of 2 CD

Solaris 9 SOFTWARE 1 of 2 と Solaris 9 SOFTWARE 2 of 2 CD からイメージをコピーした後、インストール要件に応じて、Solaris 9 INSTALLATION CD と Solaris 9 LANGUAGES CD のイメージもコピーできます。

イメージをインストールサーバーのハードディスクにコピーすると、単一のインストールサーバーから複数の Solaris リリースにディスクイメージをコピーすることもできます。

インストールサーバーの作成についての詳細は、次に示す節のどれかを参照してください。

- 71 ページの「DVD メディアを使用してインストールサーバーを作成する方法」
- 81 ページの「CD メディアを使用してインストールサーバーを作成する方法」
- ブートサーバー - オペレーティング環境をインストールするため、同一ネットワークサブネット上の複数のクライアントに対して、ブートに必要な情報を提供するサーバーシステム。ブートサーバーとインストールサーバーは、通常は同じシステムです。ただし、インストールされる Solaris 9 ソフトウェアのシステムがインストールサーバーとは異なるサブネット上に存在する場合、DHCP を使用していないときは、そのサブネットにブートサーバーが必要です。

注 - DHCP を使用する場合は、ブートサーバーを個別に作成する必要はありません。詳細は、67 ページの「DHCP によるネットワークインストールパラメータの指定」を参照してください。

ブートサーバーの作成についての詳細は、次に示す節のいずれかを参照してください。

- 74 ページの「DVD イメージを用いたサブネット上でのブートサーバーの作成」
- 86 ページの「CD イメージによるサブネット上でのブートサーバーの作成」
- (省略可能) ネームサーバー - 分散ネットワークデータベース (DNS、NIS、NIS+、LDAP など) を管理するシステム。これには、同じネットワーク上のユーザーやほかのシステムの情報が含まれています。

ネームサーバーの作成方法については、『Solaris のシステム管理 (ネーミングとディレクトリサービス : DNS、NIS、LDAP 編)』を参照してください。

注 - インストールサーバーとネームサーバーは、同じシステムであっても別のシステムであってもかまいません。

図 10-1 は、ネットワークインストールに使用される一般的なサーバー構成を示したものです。

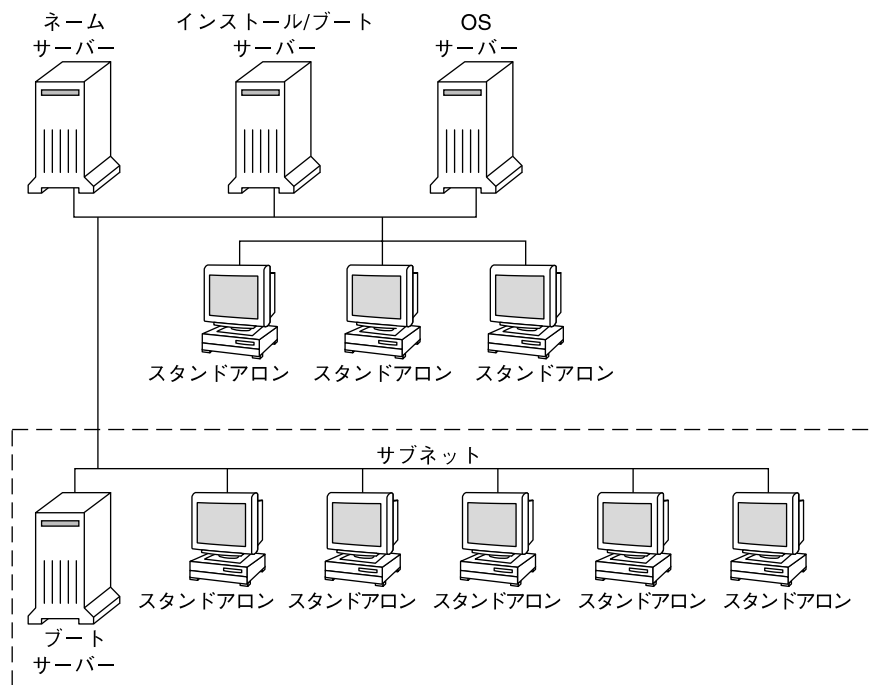


図 10-1 ネットワークインストールサーバー

DHCP によるネットワークインストールパラメータの指定

DHCP を使用する場合は、個別のブートサーバーを作成する必要はありません。インストールサーバーの作成後、`add_install_client -d` コマンドを使用してネットワークにクライアントを追加します。-d オプションを指定すると、DHCP を使用してネットワーク経由で Solaris インストールできるようにクライアントシステムを設定できます。DHCP は、インストールに必要なネットワークパラメータを提供します。インストールパラメータ用の DHCP オプションについては、『Solaris のシステム管理 (IP サービス)』の「DHCP サービスを使用した Solaris ネットワークインストールクライアントのサポート」を参照してください。

第 11 章

DVD メディアを使用してネットワークからインストールするための準備

この章では、Solaris ソフトウェアをネットワークからインストールする準備として DVD メディアを使用してネットワークとシステムを設定する方法を説明します。ネットワークインストールを使用すると、Solaris 9 ディスクイメージにアクセスできるシステム (インストールサーバーと呼ばれる) から、ネットワーク上のほかのシステムに Solaris ソフトウェアをインストールできます。その場合には、まず Solaris 9 DVD メディアの内容をインストールサーバーのハードディスクにコピーする必要があります。これで、任意の Solaris インストール方法を使用して Solaris ソフトウェアをネットワークからインストールできます。この章の内容は次のとおりです。

- 69 ページの「作業マップ: DVD メディアを使用して Solaris ソフトウェアをネットワークからインストールする準備」
- 70 ページの「DVD メディアを用いたインストールサーバーの作成」
- 74 ページの「DVD イメージを用いたサブネット上でのブートサーバーの作成」
- 76 ページの「ネットワークからインストールするシステムの追加」

作業マップ: DVD メディアを使用して Solaris ソフトウェアをネットワークからインストールする準備

表 11-1 作業マップ: DVD メディアを使用するインストールサーバーの設定

作業	説明	参照先
インストールサーバーを作成する	<code>setup_install_server(1M)</code> コマンドを使用して Solaris 9 DVD をインストールサーバーのハードディスクにコピーします。	70 ページの「DVD メディアを用いたインストールサーバーの作成」

表 11-1 作業マップ: DVD メディアを使用するインストールサーバーの設定 (続き)

作業	説明	参照先
(省略可能) ブートサーバーを作成する	インストールサーバーとは異なるサブネット上にあるシステムをネットワークからインストールする場合、システムをブートするためのブートサーバーをサブネット上に作成する必要があります。ブートサーバーを設定するには、 <code>setup_install_server</code> コマンドを <code>-b</code> オプションを付けて実行します。DHCP (Dynamic Host Configuration Protocol) を使用している場合、ブートサーバーは必要ありません。	74 ページの「DVD イメージを用いたサブネット上でのブートサーバーの作成」
ネットワークインストールするシステムを追加する	<code>add_install_client</code> コマンドを使用して、ネットワークからインストールする各システムにを設定します。インストールする各システムには、ネットワーク上にあるインストールサーバー、ブートサーバー、および構成情報を認識させる必要があります。	76 ページの「ネットワークからインストールするシステムの追加」

DVD メディアを用いたインストールサーバーの作成

インストールサーバーには、ネットワークからシステムをインストールするために必要なインストールイメージがあります。Solaris ソフトウェアをネットワークからインストールするためには、インストールサーバーを作成する必要があります。ブートサーバーは必ずしも設定する必要はありません。

- DHCP を使用してインストールパラメータを設定している場合、あるいは、インストールサーバーとクライアントが同じサブネット上にある場合、ブートサーバーは必要ありません。
- インストールサーバーとクライアントが同じサブネット上にない場合、かつ、DHCP を使用していない場合、サブネットごとに異なるブートサーバーを作成する必要があります。インストールサーバーはサブネットごとに作成できます。しかし、インストールサーバーはより多くのディスク容量を必要とします。

▼ DVD メディアを使用してインストールサーバーを作成する方法

注 – Solaris 2.3 リリースより前にリリースされた SunOS バージョンを実行しているシステムは使用できません。

注 – この手順では、システムでボリュームマネージャ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を使用せずにメディアを管理する方法については、Solaris 9 System Administrator Collection に含まれている『Solaris のシステム管理 (基本編)』を参照してください。

1. インストールサーバーとして使用するシステム上でスーパーユーザーになります。
このシステムには DVD-ROM ドライブが必要です。さらに、システムは、このサイトのネットワークに接続されネームサービスに登録されている必要があります。ネームサービスを使用する場合、システムはすでに NIS、NIS+、DNS、または LDAP のいずれかのサービスに登録されている必要があります。ネームサービスを使用しない場合は、サイトのポリシーに従ってシステムの情報を提供する必要があります。

2. システムのドライブに **Solaris 9 DVD** を挿入します。

3. **DVD** イメージをコピーするディレクトリを作成します。

```
# mkdir -p install_dir_path
```

install_dir_path DVD イメージをコピーするディレクトリを指定します。

4. マウントされたディスクの **Tools** ディレクトリに移動します。

```
# cd /cdrom/cdrom0/s0/Solaris_9/Tools
```

5. ドライブ内の **DVD** イメージをインストールサーバーのハードディスクにコピーします。

```
# ./setup_install_server install_dir_path
```

install_dir_path DVD イメージをコピーするディレクトリを指定します。

注 - `setup_install_server` コマンドは、Solaris 9 SOFTWARE ディスクイメージをコピーする十分なディスク容量があるかどうかを調べます。利用できるディスク容量を調べるには、`df -k1` コマンドを使用します。

6. 他のシステムがインストールサーバーをマウントできるように設定する必要があるかどうかを判断します。

- インストールサーバーとクライアント (インストールしているシステム) が同じサブネット上にある場合、あるいは、DHCP を使用している場合、ブートサーバーを作成する必要はありません。手順 7 に進みます。
- インストールサーバーとクライアント (インストールしているシステム) が同じサブネット上にない場合、かつ、DHCP を使用していない場合、インストールイメージをブートサーバーと共有し、NFS デーモンを起動する必要があります。次に進みます。

- a. インストールサーバーのイメージへのパスが適切に共有されていることを確認します。

```
# share | grep install_dir_path
```

```
install_dir_path          DVD イメージがコピーされているインストールイメージへのパスを指定します。
```

- インストールサーバーのディレクトリへのパスが表示される場合、かつ、`anon=0` がオプションに表示される場合、手順 7 に進みます。
 - インストールサーバーのディレクトリへのパスが表示されない場合、あるいは、`anon=0` がオプションに表示されない場合、次に進みます。
- b. インストールサーバーをブートサーバーに利用できるようにするために、このエントリを `/etc/dfs/dfstab` ファイルに追加します。

```
share -F nfs -o ro,anon=0 -d "install server directory" install_dir_path
```

- c. インストールサーバーのディレクトリへのパスが正しく共有されていることを確認します。

```
# ps ef | grep nfsd
```

- d. `nfsd` デーモンを起動し、インストールサーバーを共有します。

- `nfsd` デーモンが動作していない場合、`nfsd` デーモンを起動します。

```
# /etc/init.d/nfs.server start
```

- `nfsd` デーモンが動作している場合、インストールサーバーを共有します。

```
# shareall
```

7. ルート (`/`) ディレクトリに移動します。


```
# cd /
```

8. Solaris 9 DVD を取り出します。

9. `setup_install_server` で作成したネットインストールイメージ上のミニルート (`/install_dir_path/Solaris_9/Tools/Boot`) にあるファイルにパッチを適用するかどうかを決定します。ブートイメージに問題がある場合は、ファイルにパッチを適用する必要があります。

- パッチを適用しない場合は、このまま続けます。
- 適用する場合は、`patchadd -c` コマンドを使用して、ミニルートにあるファイルにパッチを適用します。



注意 - `patchadd -c` を使用する前には必ず、パッチの `Readme` を読むか、ご購入先におたずねください。

10. ブートサーバーを作成する必要があるかどうかを判断します。

- DHCP を使用している場合、あるいは、インストールサーバーがクライアント (インストールしているシステム) と同じサブネット上にある場合、ブートサーバーを作成する必要はありません。76 ページの「ネットワークからインストールするシステムの追加」に進みます。
- DHCP を使用していない場合、あるいは、インストールサーバーがクライアント (インストールしているシステム) と同じサブネット上にない場合、ブートサーバーを作成する必要があります。74 ページの「DVD イメージを用いたサブネット上でのブートサーバーの作成」に進みます。

例 11-1 DVD メディアを使用するインストールサーバーの作成

次に、Solaris 9 DVD をインストールサーバーの `/export/home/s9dvdsparc` ディレクトリにコピーして、インストールサーバーを作成する方法を示します。

```
# mkdir -p /export/home/s9dvdsparc
# cd /cdrom/cdrom0/s0/Solaris_9/Tools
# ./setup_install_server /export/home/s9dvdsparc
```

If you need a separate boot server, add these steps:

```
Add the following path to the /etc/dfs/dfstab file and start NFS:
share -F nfs -o ro,anon=0 -d "install server directory" /export/home/s9dvdsparc
# ps ef | grep nfsd
# /etc/init.d/nfs.server start# cd /
```

この例では、各コマンドを実行する前に、各 DVD がドライブに挿入され、自動的にマウントされるものとします。各コマンドを実行した後、DVD は取り出されます。

DVD イメージを用いたサブネット上でのブートサーバーの作成

Solaris ソフトウェアをネットワークからインストールするためには、インストールサーバーを作成する必要があります。ブートサーバーは必ずしも設定する必要はありません。ブートサーバーはシステムをネットワークからブートするために必要なブートソフトウェアを持っているので、インストールサーバーは Solaris ソフトウェアをインストールできます。

- DHCP を使用してインストールパラメータを設定している場合、あるいは、インストールサーバーとクライアントが同じサブネット上にある場合、ブートサーバーは必要ありません。76 ページの「ネットワークからインストールするシステムの追加」に進みます。
- インストールサーバーとクライアントが同じサブネット上にない場合、かつ、DHCP を使用していない場合、サブネットごとに異なるブートサーバーを作成する必要があります。インストールサーバーはサブネットごとに作成できます。しかし、インストールサーバーはより多くのディスク容量を必要とします。

▼ DVD イメージを使用してサブネット上にブートサーバーを作成する方法

1. サブネット内でブートサーバーとして使用するシステムにログインし、スーパーユーザーになります。

システムはリモートの Solaris 9 ディスクイメージ (通常は、インストールサーバー) へのアクセス権を持っている必要があります。ネームサービスを使用する場合、システムはすでにネームサービスに登録されている必要があります。ネームサービスを使用しない場合は、サイトのポリシーに従ってシステムの情報を供給する必要があります。

2. インストールサーバーから **Solaris 9 DVD** をマウントします。

```
# mount -F nfs -o ro server_name:path /mnt
```

server_name: path インストールサーバー名と、ディスクイメージへの絶対パスを指定します。

3. ブートイメージ用のディレクトリを作成します。

```
# mkdir -p boot_dir_path
```

`boot_dir_path` ブートソフトウェアをコピーするディレクトリを指定します。

4. Solaris 9 DVD イメージ上の Tools ディレクトリに移動します。

```
# cd /mnt/Solaris_9/Tools
```

5. ブートソフトウェアをブートサーバーにコピーします。

```
# ./setup_install_server -b boot_dir_path
```

`-b` システムをブートサーバーとして設定することを示します。

`boot_dir_path` ブートソフトウェアをコピーするディレクトリを指定します。

注 `-setup_install_server` コマンドは、ディスクイメージをコピーする十分なディスク容量があるかどうかを示します。利用できるディスク容量を調べるには、`df -k1` コマンドを使用します。

6. ルートディレクトリ (/) に移動します。

```
# cd /
```

7. インストールイメージをマウント解除します。

```
# umount /mnt
```

以上の操作で、ネットワークからインストールするためにシステムを設定する用意が整いました。76 ページの「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。

例 11-2 サブネット上にブートサーバーを作成する

次の例に、ブートサーバーをサブネット上に作成する方法を示します。これらのコマンドを実行すると、ブートソフトウェアが Solaris 9 DVD イメージからシステムのローカルディスク上の `/export/home/s9dvdsparc` にコピーされます。

```
# mount -F nfs -o ro crystal:/export/home/s9dvdsparc /mnt
# mkdir -p /export/home/s9dvdsparc
# cd /mnt/Solaris_9/Tools
# ./setup_install_server -b /export/home/s9dvdsparc
# cd /
# umount /mnt
```

この例では、各コマンドを実行する前に、各 DVD がドライブに挿入され、自動的にマウントされるものとしします。各コマンドを実行すると、DVD は取り出されます。

ネットワークからインストールするシステムの追加

インストールサーバーとブートサーバー (必要な場合) を作成した後、ネットワークからインストールする各システムを設定する必要があります。インストールする各システムは、次の情報にアクセスできる必要があります。

- インストールサーバー
- ブートサーバー (必要な場合)
- `sysidcfg` ファイル (`sysidcfg` ファイルを使用してシステム構成情報を事前設定する場合)
- ネームサーバー (ネームサービスを使用してシステム構成情報を事前設定する場合)
- プロファイルサーバー上の `JumpStart` ディレクトリにあるプロファイル (カスタム `JumpStart` インストールを使用する場合)

インストールサーバーとクライアントを設定するには、次の `add_install_client` 手順を使用します。また、次の手順の例も参照してください。

- DHCP を使用してインストールパラメータを設定している場合、例 11-3 を参照してください。
- インストールサーバーとクライアントが同じサブネット上にある場合、例 11-4 を参照してください。
- インストールサーバーとクライアントが同じサブネット上にない場合、かつ、DHCP を使用していない場合、例 11-5 を参照してください。

このコマンドで使用できるオプションについての詳細は、`add_install_client (1M)` のマニュアルページを参照してください。

▼ サーバーから `add_install_client` を実行してネットワークからインストールするシステムを追加する方法

ブートサーバーを持っている場合、インストールサーバーのインストールイメージを共有しており、適切なデーモンを起動していることを確認します。手順 6 を参照してください。

1. インストールサーバーまたはブートサーバー上でスーパーユーザーになります。
2. **NIS**、**NIS+**、**DNS**、または **LDAP** のネームサービスを使用する場合、インストールするシステムについての次のような情報がネームサービスに追加されていることを確認します。

- ホスト名
- IP アドレス
- Ethernet アドレス

ネームサービスについての詳細は、『Solaris のシステム管理 (ネーミングとディレクトリサービス: DNS、NIS、LDAP 編)』を参照してください。

3. インストールサーバーの **Solaris 9 DVD** イメージ上の **Tools** ディレクトリに移動します。

```
# cd /install_dir_path/Solaris_9/Tools
```

install_dir_path Tools ディレクトリへのパスを指定します。

4. ネットワークからインストールするシステムを設定します。システムはインストールサーバーにもブートサーバーにもなります。

```
# ./add_install_client [-d] [-s install_server:install_dir_path] \  
[-c jumpstart_server:jumpstart_dir_path] [-p sysid_server:path] \  
[-t boot_image_path] client_name platform_group
```

-d クライアントが DHCP を使用してネットワークインストールパラメータを取得することを指定します。

-s *install_server:install_dir_path* インストールサーバーの名前とパスを指定します。
 ■ *install_server* は、インストールサーバーのホスト名です。
 ■ *install_dir_path* は、Solaris 9 DVD イメージへの絶対パスです。

-c *jumpstart_server:jumpstart_dir_path* カスタム JumpStart インストールの JumpStart ディレクトリを指定します。 *jumpstart_server* は、JumpStart ディレクトリがあるサーバーのホスト名です。 *jumpstart_dir_path* は、JumpStart ディレクトリへの絶対パスです。

-p *sysid_server:path* システム情報を事前設定するための *sysidcfg* ファイルへのパスを指定します。 *sysid_server* は、このファイルを持っているサーバーの有効なホスト名または IP アドレスです。 *path* は、*sysidcfg* ファイルが入っているディレクトリへの絶対パスです。

-t *boot_image_path* Solaris 9 ネットインストールイメージ、CD、または DVD 上の Tools ディレクトリにあるブートイメージ以外のブートイメージを使用したい場合、別のブートイメージへのパスを指定します。

<i>client_name</i>	ネットワークからインストールするシステムのホスト名を指定します。この名前はインストールサーバーのホスト名ではありません。
<i>platform_group</i>	インストールするシステムのプラットフォームグループを指定します。詳細は、第 39 章を参照してください。

例 11-3 DHCP を使用している場合のインストールクライアントの追加方法

次の例に、DHCP を使用してネットワーク上のインストールパラメータを設定しているときに、インストールクライアントを追加する方法を示します。インストールクライアントは *basil* という名前の Ultra™ 5 システムです。ファイルシステム `/export/home/s9dvdsparc/Solaris_9/Tools` には `add_install_client` コマンドが入っています。

```
install_server# cd /export/home/s9dvdsparc/Solaris_9/Tools
install_server# ./add_install_client -d basil sun4u
```

例 11-4 インストールサーバーと同じサブネット上にあるインストールクライアントの追加方法

次の例に、インストールサーバーと同じサブネット上にあるインストールクライアントを追加する方法を示します。インストールサーバーは *basil* という名前の Ultra™ 5 システムです。ファイルシステム `/export/home/s9dvdsparc/Solaris_9/Tools` には `add_install_client` コマンドが入っています。

```
install_server# cd /export/home/s9dvdsparc/Solaris_9/Tools
install_server# ./add_install_client basil sun4u
```

例 11-5 ブートサーバーのみ存在するネットワーク上にあるインストールクライアントの追加方法

次の例に、インストールサーバーと異なるサブネット上にあるインストールクライアントを追加する方法を示します。インストールクライアントは *rose* という名前の Ultra™ 5 システムです。次のコマンドはブートサーバー上で実行します。-s オプションには、`/export/home/s9dvdsparc` に Solaris 9 DVD イメージを持っている *basil* という名前のインストールサーバーを指定しています。

```
boot_server# cd /export/home/s9dvdsparc/Solaris_9/Tools
boot_server# ./add_install_client -s basil:/export/home/s9dvdsparc rose sun4u
```

第 12 章

CD メディアを使用してネットワークからインストールするための準備

この章では、ネットワーク上で Solaris ソフトウェアをインストールする準備として CD メディアを使用してネットワークとシステムを設定する方法を説明します。ネットワークインストールを使用すると、Solaris 9 CD イメージまたは DVD イメージにアクセスできるシステム (インストールサーバーと呼ばれる) から、ネットワーク上のほかのシステムに Solaris ソフトウェアをインストールできます。ユーザーは、まず CD メディアの内容をインストールサーバーのハードディスクにコピーする必要があります。これで、任意の Solaris インストール方法を使用して Solaris ソフトウェアをネットワーク上でインストールできます。この章の内容は次のとおりです。

- 80 ページの「作業マップ: CD メディアを使用してネットワーク上でインストールするための準備」
- 81 ページの「CD メディアによるインストールサーバーの作成」
- 86 ページの「CD イメージによるサブネット上でのブートサーバーの作成」
- 88 ページの「ネットワークからインストールするシステムの追加」

作業マップ: CD メディアを使用してネットワーク上でインストールするための準備

表 12-1 作業マップ: CD メディアによるインストールサーバーの設定

作業	説明	参照先
インストールサーバーを作成する	<p>setup_install_server(1M) コマンドを使用して Solaris 9 SOFTWARE 1 of 2 CD をインストールサーバーのハードディスクにコピーします。</p> <p>次に、add_to_install_server(1M) コマンドを使用して Solaris 9 SOFTWARE 2 of 2 CD と Solaris 9 LANGUAGES CD をインストールサーバーのハードディスクに追加します。</p> <p>さらに、modify_install_server(1M) コマンドを使用してネットワークインストールイメージに Solaris Web Start ユーザーインタフェースソフトウェアを追加します。</p>	81 ページの「CD メディアによるインストールサーバーの作成」
(省略可能) ブートサーバーを作成する	インストールサーバーと異なるサブネット上にあるシステムをネットワーク上でインストールする場合は、システムをブートするためのブートサーバーをサブネット上に作成する必要があります。DHCP (Dynamic Host Configuration Protocol) を使用する場合は、ブートサーバーは不要です。	86 ページの「CD イメージによるサブネット上でのブートサーバーの作成」
ネットワークインストールするシステムを追加する	add_install_client コマンドを使用して、ネットワーク上でインストールする各システムを設定します。インストールする各システムは、ネットワーク上でインストールサーバー、ブートサーバー、および構成情報を認識する必要があります。	88 ページの「ネットワークからインストールするシステムの追加」

CD メディアによるインストールサーバーの作成

インストールサーバーには、ネットワーク上でシステムをインストールするために必要なインストールイメージが含まれます。ネットワーク上で Solaris ソフトウェアをインストールするためには、インストールサーバーを作成する必要があります。必ずしも個別のブートサーバーを設定する必要はありません。

- DHCP を使用してインストールパラメータを設定する場合や、インストールサーバーとクライアントが同じサブネット上に存在する場合は、個別のブートサーバーは不要です。
- インストールサーバーとクライアントが同じサブネット上に存在せず、DHCP を使用していない場合は、サブネットごとに個別のブートサーバーを作成する必要があります。インストールサーバーはサブネットごとに作成できますが、インストールサーバーにはより多くのディスク容量が必要です。

▼ CD メディアを使用してインストールサーバーを作成する方法

注 – Solaris 2.3 以前にリリースされた SunOS バージョンを使用するシステムは使用できません。

注 – この手順では、システムでボリュームマネージャ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を使用せずにメディアを管理する方法については、Solaris 9 Beta System Administrator Collection に含まれている『Solaris のシステム管理 (基本編)』を参照してください。

1. インストールサーバーとして使用するシステムでスーパーユーザーになります。このシステムには **CD-ROM** ドライブが必要です。
さらに、システムは、このサイトのネットワークに接続されネームサービスに登録されている必要があります。ネームサービスを使用する場合は、システムがすでに NIS、NIS+、DNS、LDAP のいずれかのネームサービスに登録されていなければなりません。ネームサービスを使用しない場合は、サイトのポリシーに従ってシステムの情報を供給する必要があります。
2. **Solaris 9 SOFTWARE 1 of 2 CD** をシステムのドライブに挿入します。

3. CD イメージ用のディレクトリを作成します。

```
# mkdir -p install_dir_path
```

install_dir_path CD イメージをコピーするディレクトリを指定します。

4. マウントされたディスクの Tools ディレクトリに移動します。

```
# cd /cdrom/cdrom0/s0/Solaris_9/Tools
```

5. ドライブ内のイメージをインストールサーバーのハードディスクにコピーします。

```
# ./setup_install_server install_dir_path
```

install_dir_path CD イメージをコピーするディレクトリを指定します。

注 - `setup_install_server` コマンドは、Solaris 9 ソフトウェアディスクイメージをコピーする十分なディスク容量があるかどうかを示します。利用できるディスク容量を調べるには、`df -k1` コマンドを使用します。

6. 他のシステムがインストールサーバーをマウントできるように設定する必要があるかどうかを判断します。

- インストールサーバーがインストールするシステムと同じサブネット上にあるか、あるいは DHCP を使用している場合は、ブートサーバーを作成する必要はありません。手順 7 へ進みます。
- インストール対象のシステムと同じサブネット上にインストールサーバーが存在せず、DHCP を使用していないという場合は、ブートサーバーとインストールイメージを共有し、NFS デーモンを起動する必要があります。次の手順へ進みます。
- a. インストールサーバーのインストールイメージのパスが適切に共有されているかを確認します。

```
# share | grep install_dir_path
```

install_dir_path ブートサーバーのインストールイメージのパスを指定します。

- インストールサーバーのディレクトリのパスが表示されてオプションに `anon=0` と示された場合は、手順 7 へ進みます。
- インストールサーバーのディレクトリのパスが表示されないか、オプション内に `anon=0` と示されない場合は、次の作業へ進みます。

- b. 次のエントリを `/etc/dfs/dfstab` ファイルに追加し、ブートサーバーからインストールサーバーが利用できるように設定します。

```
share -F nfs -o ro,anon=0 -d "install server directory" install_dir_path
```

- c. インストールサーバーのディレクトリパスが正しく共有されているか確認します。

```
# ps ef | grep nfsd
```

- d. `nfsd` デーモンを起動し、インストールサーバーを共有します。

- `nfsd` デーモンが稼動していない場合は、このデーモンを起動してください。

```
# /etc/init.d/nfs.server start
```

- `nfsd` がすでに稼動している場合は、インストールサーバーを共有してください。

```
# shareall
```

7. `root (/)` ディレクトリへ移動します。

```
# cd /
```

8. **Solaris 9 SOFTWARE 1 of 2 CD** を取り出します。

9. **Solaris 9 SOFTWARE 2 of 2 CD** をシステムの **CD-ROM** ドライブに挿入します。

10. マウントされた **CD** の `Tools` ディレクトリに移動します。

```
# cd /cdrom/cdrom0/Solaris_9/Tools
```

11. **CD-ROM** ドライブ内の **CD** をインストールサーバーのハードディスクにコピーします。

```
# ./add_to_install_server install_dir_path
```

install_dir_path CD イメージをコピーするディレクトリを指定します。

12. `root (/)` ディレクトリへ移動します。

```
# cd /
```

13. **Solaris 9 SOFTWARE 2 of 2 CD** を取り出します。

14. **Solaris 9 LANGUAGES CD** をシステムの **CD-ROM** ドライブに挿入します。

15. マウントされた **CD** の `Tools` ディレクトリに移動します。

```
# cd /cdrom/cdrom0/sol_9_lang_sparc/Tools
```

16. **CD-ROM** ドライブ内の **CD** をインストールサーバーのハードディスクにコピーします。

```
# ./add_to_install_server install_dir_path
```

install_dir_path CD イメージをコピーするディレクトリを指定します。

17. **root (/)** ディレクトリへ移動します。

```
# cd /
```

18. ユーザーが **Solaris Web Start** を使用してシステムをブートし、ネットワーク上で **Solaris 9** ソフトウェアをインストールできるように

- しない場合は、Solaris 9 LANGUAGES CD を取り出して手順 22 へ進みます。
- する場合は、Solaris 9 LANGUAGES CD を取り出して続きます。

19. **Solaris 9 INSTALLATION CD** をシステムの **CD-ROM** ドライブに挿入します。

20. マウントされた **CD** の *modify_install_server* を含むディレクトリに移動します。

```
# cd /cdrom/cdrom0/s0
```

21. **Solaris Web Start** インタフェースソフトウェアをインストールサーバーにコピーします。

```
# ./modify_install_server -p install_dir_path installer_miniroot_path
```

-p 既存イメージのミニルート *install_dir_path/Solaris_9/Tools/Boot.orig* に保存します。

install_dir_path Solaris Web Start インストールをコピーするディレクトリを指定します。

installer_miniroot_path Solaris Web Start インタフェースのコピー元となる CD 上のディレクトリを指定します。

22. *setup_install_server* によって作成されたネットインストールイメージ上のミニルート (*/install_dir_path/Solaris_9/Tools/Boot*) にあるファイルにパッチを適用するか決定します。ブートイメージに何か問題がある場合は、ファイルへのパッチの適用が欠かせない可能性があります。

- パッチを適用しない場合は、そのまま続きます。
- 適用する場合は、`patchadd -c` コマンドを使用して、ミニルートにあるファイルにパッチを適用します。



注意 `- patchadd -c` は、『パッチの **Readme**』の指示に目を通すか、あるいはご購入先に連絡をとった上で使用してください。

23. ブートサーバーを作成する必要があるかどうかを判断します。

- DHCP を使用しているか、あるいはインストール対象のシステムと同じサブネット上にインストールサーバーが存在する場合は、ブートサーバーを作成する必要はありません。76 ページの「ネットワークからインストールするシステムの追加」へ進みます。
- DHCP を使用しておらず、インストールサーバーとクライアントが別のサブネットに存在する場合は、ブートサーバーを作成する必要があります。74 ページの「DVD イメージを用いたサブネット上でのブートサーバーの作成」へ進みます。

例 12-1 CD メディアによるインストールサーバーの作成

次の例は、インストールサーバーの `/export/home/s9cdsparc` ディレクトリに次の CD をコピーしてインストールサーバーを作成する方法を示しています。

- Solaris 9 SOFTWARE 1 of 2 CD (SPARC)
- Solaris 9 SOFTWARE 2 of 2 CD (SPARC)
- Solaris 9 LANGUAGES CD (SPARC)

```
# mkdir -p /export/home/s9cdsparc
# cd /cdrom/cdrom0/s0/Solaris_9/Tools
# ./setup_install_server /export/home/s9cdsparc
```

インストールサーバーが異なる場合、次の手順を追加。

次のパスを `/etc/dfs/dfstab` ファイルに追加。

```
share -F nfs -o ro,anon=0 -d "install server directory" \
/export/home/s9cdsparc
# ps ef | grep nfsd
# /etc/init.d/nfs.server start
```

次の手順を行う。

ブートサーバーが必要な場合は、次の手順を行う。

```
# cd /
# cd /cdrom/cdrom0/Solaris_9/Tools
# ./add_to_install_server /export/home/s9cdsparc
# cd /
# cd /cdrom/cdrom0/sol_9_lang_sparc/Tools
# ./add_to_install_server /export/home/s9cdsparc
# cd /
# cd /cdrom/cdrom0/s0
# ./modify_install_server -p /export/home/s9cdsparc /cdrom/cdrom0/s0
```

この例では、各コマンドを実行する前に各 CD がドライブに挿入され、自動的にマウントされるものとします。各コマンドを実行すると、CD が取り出されます。

CD イメージによるサブネット上での ブートサーバーの作成

ネットワーク上で Solaris ソフトウェアをインストールするためには、インストールサーバーを作成する必要があります。ブートサーバーは必ず設定する必要があるわけではありません。ブートサーバーには、ネットワーク上でシステムをブートするために必要なブートソフトウェアが含まれています。一方、インストールサーバーは Solaris ソフトウェアのインストールを行います。

- DHCP を使用してインストールパラメータを設定しているか、あるいはインストールサーバーまたはクライアントがインストールサーバーと同じサブネット上に存在する場合は、ブートサーバーは不要です。88 ページの「ネットワークからインストールするシステムの追加」へ進みます。
- インストールサーバーとクライアントが同じサブネット上に存在せず、DHCP を使用していない場合は、サブネットごとに個別のブートサーバーを作成する必要があります。インストールサーバーはサブネットごとに作成できますが、インストールサーバーにはより多くのディスク容量が必要です。

▼ CD イメージを使用してサブネット上にブートサーバーを作成する方法

1. サブネット内でブートサーバーとして使用するシステムにログインし、スーパーユーザーになります。

システムに CD-ROM ドライブが存在するか、あるいはリモートの Solaris 9 イメージ (通常、インストールサーバーに存在する) にアクセスできなければなりません。ネームサービスを使用する場合は、システムがすでにそのネームサービスに登録されていなければなりません。ネームサービスを使用しない場合は、サイトのポリシーに従ってシステムの情報を供給する必要があります。

2. インストールサーバーから **Solaris 9 SOFTWARE 1 of 2 CD** イメージをマウントします。

```
# mount -F nfs -o ro server_name:path /mnt
```

`server_name: path` インストールサーバー名とディスクイメージの絶対パスを指定します。

3. ブートイメージ用のディレクトリを作成します。

```
# mkdir -p boot_dir_path
```

`boot_dir_path` ブートソフトウェアをコピーするディレクトリを指定します。

4. **Solaris 9 SOFTWARE 1 of 2 CD** イメージの `Tools` ディレクトリへ移動します。

```
# cd /mnt/Solaris_9/Tools
```

5. ブートソフトウェアをブートサーバーにコピーします。

```
# ./setup_install_server -b boot_dir_path
```

`-b` システムをブートサーバーとして設定することを示します。

`boot_dir_path` ブートソフトウェアをコピーするディレクトリを指定します。

注 `setup_install_server` コマンドは、Solaris 9 ソフトウェアのディスクイメージをコピーする十分なディスク容量があるかどうかを調べます。利用できるディスク容量を調べるには、`df -k1` コマンドを使用します。

6. `root (/)` ディレクトリへ移動します。

```
# cd /
```

以上の操作で、ネットワーク上でインストールするためにシステムを設定する用意が整います。88 ページの「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。

例 12-2 CD メディアを使用してサブネット上にブートサーバーを作成する

次の例は、ブートサーバーをサブネット上に作成する方法を示したものです。次のコマンドを実行すると、ブートソフトウェアが Solaris 9 SOFTWARE 1 of 2 CD イメージからシステムのローカルディスク (`/export/install/boot`) にコピーされます。また、インストールサーバーに Solaris Web Start インタフェースソフトウェアがコピーされます。

```
# mount -F nfs -o ro crystal:/export/install/boot /mnt
# mkdir -p boot_dir_path
# cd /mnt/Solaris_9/Tools
# ./setup_install_server -b /export/install/boot
# cd /
```

この例では、コマンドを実行する前に各ディスクがドライブに挿入され、自動的にマウントされるものとします。コマンドを実行すると、ディスクが取り出されます。

ネットワークからインストールするシステムの追加

インストールサーバーとブートサーバー (必要な場合) を作成したあと、ネットワーク上でインストールする各システムを設定する必要があります。インストールする各システムは、次の情報にアクセスできる必要があります。

- インストールサーバー
- ブートサーバー (必要な場合)
- `sysidcfg` ファイル (`sysidcfg` ファイルを使用してシステム構成情報を事前設定する場合)
- ネームサーバー (ネームサービスを使用してシステム構成情報を事前設定する場合)
- プロファイルサーバー上の `JumpStart` ディレクトリにあるプロファイル (カスタム `JumpStart` インストールを使用する場合)

インストールサーバーとクライアントの設定には、`add_install_client` を使用して以下の作業を行なってください。作業例も併せて参照してください。

- DHCP を使用してインストールパラメータを設定する場合は、例 12-3 を参照してください。
- インストールサーバーとクライアントが同じサブネット上に存在する場合は、例 12-4 を参照してください。
- インストールサーバーとクライアントが同じサブネット上に存在せず、DHCP を使用していない場合は、例 12-5 を参照してください。

▼ サーバーから `add_install_client` を実行してネットワークからインストールするシステムを追加する方法

ブートサーバーが存在する場合は、インストールサーバーのインストールイメージを共有しているか確認してください。手順 6 を参照してください。

1. インストールサーバーまたはブートサーバー上でスーパーユーザーになります。
2. **NIS**、**NIS+**、**DNS**、または **LDAP** ネームサービスを使用する場合は、インストール対象のシステムに関する次の情報がネームサービスに追加されているか確認します。
 - ホスト名
 - IP アドレス
 - Ethernet アドレス

ネームサービスの詳細は、『Solaris のシステム管理 (ネーミングとディレクトリサービス : DNS、NIS、LDAP 編)』を参照してください。

3. インストールサーバー上の **Solaris 9 CD** イメージの **Tools** ディレクトリに移動します。

```
# cd /install_dir_path/Solaris_9/Tools
```

install_dir_path Tools ディレクトリのパスを指定します。

4. ネットワークでインストールするためのシステムを設定します。このシステムは、インストールサーバーでもブートサーバーでもかまいません。

```
# ./add_install_client [-d] [-s install_server:install_dir_path] \  
[-c jumpstart_server:jumpstart_dir_path] [-p server:path] \  
[-t boot_image_path] client_name platform_group
```

-d クライアントが DHCP を使用してネットワークインストールパラメータを取得することを指定します。

-s *install_server:install_dir_path* インストールサーバーの名前とパスを指定します。
■ *install_server* は、インストールサーバーのホスト名です。
■ *install_dir_path* は、Solaris 9 CD イメージの絶対パスです。

-c *jumpstart_server:jumpstart_dir_path* カスタム JumpStart インストールの JumpStart ディレクトリを指定します。*jumpstart_server* には JumpStart ディレクトリがあるサーバーのホスト名を指定します。*jumpstart_dir_path* には JumpStart ディレクトリの絶対パスを指定します。

-p *sysid_server:path* システムの構成情報を事前設定するための *sysidcfg* ファイルのパスを指定します。*sysid_server* は、このファイルを含んでいるサーバーの有効なホスト名または IP アドレスです。*path* は、*sysidcfg* ファイルを含んでいるディレクトリの絶対パスです。

-t *boot_image_path* Solaris 9 ネットインストールイメージ、CD、または DVD 上の Tools ディレクトリ内のものとは別のブートイメージを使用したい場合は、代わりとなるブートイメージのパスを指定します。

client_name ネットワーク上でインストールされるシステムのホスト名 (インストールサーバーのホスト名ではない)。

platform_group

インストールするシステムのプラットフォームグループ。プラットフォームグループの詳細は、第 39 章を参照してください。

例 12-3 DHCP を使用している場合のインストールサーバー上のインストールクライアントの追加方法 (CD)

次の例に、DHCP を使用してネットワーク上でインストールパラメータを設定しているときに、インストールクライアントを追加する方法を示します。インストールクライアントは、*basil* という名前の Ultra™ 5 システムです。ファイルシステム `/export/home/s9cdsparc/Solaris_9/Tools` には、`add_install_client` コマンドが入っています。

```
install_server# cd /export/home/s9cdsparc/Solaris_9/Tools
install_server# ./add_install_client -d basil sun4u
```

例 12-4 サーバーと同じサブネット上にあるインストールクライアントの追加方法 (CD)

次の例に、インストールサーバーと同じサブネット上にあるインストールクライアントを追加する方法を示します。インストールサーバーは、*basil* という名前の Ultra™ 5 システムです。ファイルシステム `/export/home/s9cdsparc/Solaris_9/Tools` には、`add_install_client` コマンドが入っています。

```
install_server# cd /export/home/s9cdsparc/Solaris_9/Tools
install_server# ./add_install_client basil sun4u
```

例 12-5 ブートサーバーのみ存在するネットワーク上にあるインストールクライアントの追加方法 (CD)

次の例に、インストールサーバーと異なるサブネット上にあるインストールクライアントを追加する方法を示します。インストールクライアントは、*rose* という名前の Ultra™ 5 システムです。次のコマンドはブートサーバー上で実行します。-s オプションには、`/export/home/s9cdsparc` に Solaris 9 CD イメージを持っている *basil* という名前のインストールサーバーを指定しています。

```
boot_server# cd /export/home/s9cdsparc/Solaris_9/Tools
boot_server# ./add_install_client -s basil:/export/home/s9cdsparc rose sun4u
```

第 13 章

ネットワークからインストールするための準備 (コマンドリファレンス)

この章では、ネットワークインストールを設定するときに使用できるコマンドを示します。

表 13-1 ネットワークインストールコマンド

コマンド	説明
<code>add_install_client</code>	システムがネットワークからインストールできるように、必要なネットワークインストール情報をインストールサーバーまたはブートサーバーに追加するコマンド。詳細は、 <code>add_install_client(1M)</code> を参照
<code>setup_install_server</code>	Solaris 9 DVD または CD をインストールサーバーのローカルディスクにコピーするか、ブートソフトウェアをブートサーバーにコピーするスクリプト。詳細は、 <code>setup_install_server(1M)</code> のマニュアルページを参照
<code>add_to_install_server</code> (CD メディアのみ)	CD 上のプロダクトツリー内の追加パッケージを、既存のインストールサーバー上のローカルディスクにコピーするスクリプト。詳細は、 <code>add_to_install_server(1M)</code> のマニュアルページを参照
<code>modify_install_server</code> (CD メディアのみ)	Solaris Web Start ユーザーインタフェースソフトウェアを、既存のブートサーバー上の Solaris 9 CD イメージに追加するスクリプト。ユーザーが Solaris Web Start によってシステムをブートし、Solaris 9 ソフトウェアをネットワークからインストールできるようにする。詳細は、 <code>modify_install_server(1M)</code> のマニュアルページを参照RAM が 64M バイト以下のシステムは、Solaris Web Start を使用してマシンをインストールできません。
<code>mount</code>	ファイルシステムをマウントできるようにし、マウントされたファイルシステム (Solaris 9 DVD または Solaris 9 SOFTWARE と Solaris 9 LANGUAGES CD 上のファイルシステムを含む) を表示するコマンド。詳細は、 <code>mount(1M)</code> のマニュアルページを参照

表 13-1 ネットワークインストールコマンド (続き)

コマンド	説明
uname -i	システムのプラットフォーム名 (SUNW、SPARCstation-5 など) を調べるコマンド。Solaris ソフトウェアをインストールするには、システムのプラットフォーム名が必要になることがあります。詳細は、uname (1) のマニュアルページを参照
patchadd -C <i>net_install_image</i>	<p>setup_install_server で作成した DVD または CD のネットワークインストールイメージ上にあるミニルート (Solaris_9/Tools/Boot) にあるファイルにパッチを追加するコマンド。これにより、Solaris インストールコマンドとほかのミニルート固有のコマンドにパッチを適用できる。net_install_image はネットワークインストールイメージの絶対パス名である。詳細は、patchadd (1M) のマニュアルページを参照</p> <p>注意 - patchadd -c を使用する前には必ず、パッチの Readme を読むか、ご購入先におたずねください。</p>
reset	システムをリセットし、マシンを再起動するための Open Boot prom コマンド。ブート時に入出力割り込みに関するエラーメッセージが表示された場合は、L1 または STOP キーと A キーを同時に押し、その後 ok または >PROM プロンプトで reset と入力する。
banner	モデル名、Ethernet アドレス、インストールされているメモリーなどのシステム情報を表示する Open Boot prom コマンド。ok または PROM プロンプトでのみ使用可能。

第 14 章

Solaris Web Start の使用

この章では、Solaris 9 DVD または Solaris 9 INSTALLATION CD に含まれている Solaris Web Start を使用して、Solaris ソフトウェアのインストールやアップグレードを行う方法について説明します。

注 – Solaris オペレーティング環境をインストールするシステムまたはドメインに直接接続された DVD-ROM ドライブまたは CD-ROM ドライブがない場合は、別のマシンに接続された DVD-ROM ドライブまたは CD-ROM ドライブを使用できます。詳細は、付録 B を参照してください。

この章の内容は、次のとおりです。

- 93 ページの「Solaris Web Start の実行方法」
- 94 ページの「Solaris Web Start によるインストールまたはアップグレード」
- 98 ページの「Solaris Web Start によるインストールまたはアップグレード後に必要な作業」

Solaris Web Start の実行方法

Solaris Web Start は、GUI または CLI を使って実行できます。

- GUI – ローカルまたはリモートの DVD-ROM ドライブか CD-ROM ドライブ、またはネットワーク接続、およびビデオアダプタ、キーボード、モニター、十分なメモリーが必要です。
- CLI – ローカルまたはリモートの DVD-ROM ドライブか CD-ROM ドライブ、またはネットワーク接続、およびキーボードとモニターが必要です。Solaris Web Start インストール CLI は、`tip` コマンドを使って実行できます。詳細は、`tip(1)` のマニュアルページを参照してください。

Solaris Web Start は、システムにビデオアダプタがあることを検出すると自動的に GUI を使用し、検出できないと CLI を使用します。操作手順や設定内容は、GUI も CLI もほとんど同じです。

Solaris Web Start によるインストールまたはアップグレード

Solaris 9 DVD または Solaris 9 INSTALLATION CD に含まれている Solaris Web Start プログラムを使用して、Solaris オペレーティング環境をインストールまたはアップグレードできます。

注 – Solaris 9 INSTALLATION CD からインストールする場合は、25 ページの「Solaris 9 INSTALLATION CD を使用するための要件」で特殊な要件を確認してください。

以下のものが揃っているか確認します。

- DVD からインストールする場合は、Solaris 9 DVD を使用してください。
- CD からインストールする場合は、以下の CD を使用してください。
 - Solaris 9 INSTALLATION CD
 - Solaris 9 SOFTWARE 1 of 2 CD
 - Solaris 9 SOFTWARE 2 of 2 CD – 必要に応じて、インストールプログラムがこの CD を挿入するように要求します。
 - Solaris 9 LANGUAGES CD – 特定地域の言語をサポートするために必要な場合、インストールプログラムがこの CD を挿入するように要求します。

作業マップ: Solaris Web Start インストールの実行

表 14-1 作業マップ: Solaris Web Start インストールの実行

作業	説明	参照先
システム要件を確認する	Solaris Web Start を使用してインストールまたはアップグレードをするための要件をシステムが満たしていることを確認します。	24 ページの「システム要件」

表 14-1 作業マップ: Solaris Web Start インストールの実行 (続き)

作業	説明	参照先
必要な情報を収集する	チェックリストに従ってワークシートに記入しながら、Solaris ソフトウェアのインストールに必要なすべての情報を収集します。	第 6 章
(省略可能) システム構成情報を事前設定する	sysidcfg ファイルやネームサービスを使って、システムのインストール情報(locale など)を事前設定することができます。これによって、インストール時にこの情報を入力しなくて済みます。	第 7 章
(アップグレードのみ) システムをアップグレードするための準備をする	システムのバックアップをとります。	『Solaris のシステム管理(基本編)』
(省略可能) ネットワークからインストールするためのシステムを設定する	リモートの DVD または CD ネットインストールイメージをインストールする場合は、インストールサーバーまたはブートサーバーからシステムをブートしてインストールするように、システムを設定する必要があります。	第 12 章
インストールまたはアップグレードを行う	システムをブートし、プロンプトに従って Solaris ソフトウェアをインストールまたはアップグレードします。	95 ページの「Solaris Web Start を使用してインストールまたはアップグレードする方法」
(アップグレードのみ) アップグレード後に必要な作業を行う	アップグレードで失われたローカル変更がある場合は、それを復元します。	98 ページの「アップグレード後にローカル変更を復元する方法」

▼ Solaris Web Start を使用してインストールまたはアップグレードする方法

1. **DVD-ROM** ドライブ、**CD-ROM** ドライブ、ネットイメージのどれを使用してソフトウェアをインストールするかを決定します。
 - DVD-ROM ドライブか CD-ROM ドライブを使用する場合は、Solaris 9 DVD または Solaris 9 INSTALLATION CD を挿入します。
 - ネットインストールイメージを使用する場合は、インストールメディアが置かれているディレクトリに移動します。必要に応じ、ネットワーク管理者に場所を尋ねてください。次にコマンド例を示します。

```
% cd /net/install-svr/export/s9/sparc
```
2. システムをブートします。
 - 新しく購入したばかり(未使用)のシステムの場合は、システムの電源を入れます。
 - 既存システムのインストールまたはアップグレードを行う場合は、システムをシャットダウンします。



図 14-1 Solaris Web Start Kiosk メニュー

Kiosk メニューのリンクは、どれでもクリックできます。

注 - Kiosk によって画面が隠れてしまうことがあります。隠れている画面を表示するには、Kiosk メニューの「Kiosk を奥へ (Send Kiosk to Background)」をクリックします。

「インストーラ・オプション (Installer Questions)」画面が表示されます。

4. システムのリポートとディスクの取り出しを自動的に行うかどうかを決定します。
「次へ (NEXT)」をクリックします。
「媒体の指定 (Specify Media)」画面が表示されます。
5. どのメディア (CD、DVD、ネットワーク、HTTP、FTP、またはローカルテープ) をインストールに使用するかを指定します。
6. 初期インストールまたはアップグレードのいずれかを選択します。

システムがアップグレード可能かどうか判定されます。このためには、Solaris ルート(/) ファイルシステムがすでに存在していなければなりません。Solaris 9 INSTALLATION CD を使用してアップグレードする場合は、512M バイトのスライスが必要です。アップグレードが可能であることが確認されると、システムのアップグレードが開始されます。

7. 画面に示される手順に従って、**Solaris** ソフトウェアと追加ソフトウェア (必要な場合) をシステムにインストールします。

Solaris ソフトウェアのインストールが終了すると、システムのリブートを求める画面が表示されます。

インストールが終了すると、インストールログがファイルに保存されます。インストールログは、次のディレクトリに作成されます。

- /var/sadm/system/logs
- /var/sadm/install/logs

8. アップグレードを行なった場合は、保存されなかったローカル変更があればそれを復元します。詳細は、98 ページの「アップグレード後にローカル変更を復元する方法」を参照してください。

Solaris Web Start によるインストールまたはアップグレード後に必要な作業

Solaris オペレーティング環境のインストールまたはアップグレードを終了した後、必要に応じて次の作業を行う必要があります。

▼ アップグレード後にローカル変更を復元する方法

アップグレードを行うと、Solaris Web Start は既存システムに対して行なったローカルなソフトウェア変更と新しい Solaris ソフトウェアをマージします。ただし場合によっては、マージが行われないことがあります。したがって、システムのアップグレードが終了したら、保存されなかったローカル変更を復元しなければならない場合があります。



注意 – 保存されなかったローカル変更を復元しないと、システムがブートされないことがあります。

1. 次のファイルの内容を確認して、**Solaris Web Start** によって保存されなかったローカル変更を復元する必要があるかどうかを判断します。

```
/a/var/sadm/system/data/upgrade_cleanup
```

2. 保存されなかったローカル変更があれば、復元します。
3. リブートします。

```
# reboot
```


第 15 章

Solaris suninstall プログラムの使用

この章では、Solaris 9 SOFTWARE 1 of 2 CD に含まれている Solaris suninstall プログラムを使用して Solaris ソフトウェアをインストールまたはアップグレードする方法について説明します。

- 101 ページの「Solaris suninstall プログラム」
- 102 ページの「Solaris suninstall プログラムによるインストールまたはアップグレード」
- 104 ページの「Solaris suninstall プログラムによるアップグレード後に必要な作業」

Solaris suninstall プログラム

Solaris suninstall プログラムは CLI で実行できます。Solaris suninstall プログラムには、ローカルまたはリモートの CD-ROM ドライブまたはネットワーク接続、キーボード、およびモニターが必要です。Solaris suninstall プログラムは `tip` コマンドで実行できます。詳細は、`tip(1)` のマニュアルページを参照してください。

注 - テキストを操作するとき、Control-U を使用するとページを上移動し、Control-D を使用するとページを下移動します。

Solaris suninstall プログラムによるインストールまたはアップグレード

Solaris 9 SOFTWARE 1 of 2 CD に含まれている Solaris suninstall プログラムを使用すると、Solaris オペレーティング環境をシステムにインストールまたはアップグレードできます。Solaris suninstall プログラムでは、追加ソフトウェアをインストールできません。

次の CD を用意してください

- Solaris 9 SOFTWARE 1 of 2 CD
- Solaris 9 SOFTWARE 2 of 2 CD – 必要に応じて、インストールプログラムがこの CD をドライブに挿入するように要求します。
- Solaris 9 LANGUAGES CD – 特定地域の言語をサポートするために必要な場合、インストールプログラムがこの CD をドライブに挿入するように要求します。

作業マップ: Solaris suninstall プログラムを実行する

表 15-1 作業マップ: Solaris suninstall プログラムを実行する

作業	説明	参照先
必要な情報を収集する	チェックリストに従ってワークシートに記入しながら、Solaris ソフトウェアのインストールに必要なすべての情報を収集します。	第 6 章
(アップグレードのみ) システムをアップグレードするための準備をする	システムのバックアップをとります。	『Solaris のシステム管理 (基本編)』
(省略可能) システム構成情報を事前設定する	sysidcfg ファイルまたはネームサービスを使用して、システムのインストール情報 (locale など) を事前設定します。システム情報をあらかじめ設定すると、インストール時に情報入力を求めるプロンプトは表示されません。	第 7 章
(省略可能) ネットワークからインストールするためのシステムを設定する	リモートの Solaris 9 SOFTWARE イメージからシステムをインストールする場合、インストールサーバーまたはブートサーバーからブートおよびインストールするようにシステムを設定する必要があります。	第 12 章

表 15-1 作業マップ: Solaris suninstall プログラムを実行する (続き)

作業	説明	参照先
インストールまたはアップグレードを行う	システムをブートし、プロンプトに従って Solaris ソフトウェアをインストールまたはアップグレードします。	103 ページの「Solaris suninstall プログラムを使用してインストールまたはアップグレードするには」
(アップグレードのみ) アップグレード後に必要な作業を行う	アップグレードで失われたローカル変更がある場合は、それを復元します。	104 ページの「Solaris suninstall プログラムによるアップグレード後に必要な作業」

▼ Solaris suninstall プログラムを使用してインストールまたはアップグレードするには

1. **CD-ROM** ドライブとネットイメージのどちらを使用してソフトウェアをインストールするかを決定します。

- CD-ROM ドライブを使用する場合、Solaris 9 SOFTWARE 1 of 2 CD を挿入します。
- ネットインストールイメージを使用する場合、次の例のように、インストールメディアが入っているディレクトリに移動します。必要に応じて、ネットワーク管理者に場所をたずねてください。次にコマンド例を示します。

```
% cd /net/install-svr/export/s9/sparc
```

2. システムをブートします。

- 新しく購入したばかり (未使用) のシステムの場合は、システムの電源を入れます。
- 既存システムのインストールまたはアップグレードを行う場合は、システムをシャットダウンします。
 - ローカル CD からブートする場合は、次のように入力します。

```
ok boot cdrom
```

- ネットワーク上のインストールサーバーからブートする場合は、次のように入力します。

```
ok boot net
```

Solaris suninstall プログラムが始まります。

3. システム構成の質問に答えます。

すべてのシステム構成情報が事前設定されている場合は、構成情報の入力は求められません。

システム構成情報の事前設定を行っていない場合は、38 ページの「インストール用のワークシート」または 43 ページの「アップグレード用のワークシート」を参照して、システム構成の質問に答えます。

4. 画面に示される手順に従って、**Solaris** ソフトウェアをシステムにインストールします。
Solaris ソフトウェアのインストールが終了すると、システムは自動的にリブートするか、または手動でリブートするようにプロンプトが表示されます。
インストールが終了すると、インストールログがファイルに保存されます。インストールログは、次のディレクトリに作成されます。
 - /var/sadm/system/logs
 - /var/sadm/install/logs
5. アップグレードを行なった場合は、保存されなかったローカル変更があればそれを復元します。詳細は、104 ページの「**Solaris suninstall** プログラムによるアップグレード後に必要な作業」を参照してください。
6. (省略可能) 追加のソフトウェアをインストールする方法は、『Solaris のシステム管理 (基本編)』を参照してください。

Solaris suninstall プログラムによる アップグレード後に必要な作業

アップグレードを行うと、Solaris suninstall プログラムは、既存システムに対して行なったローカルなソフトウェア変更と新しい Solaris ソフトウェアをマージします。ただし場合によっては、マージが行われないことがあります。したがって、システムのアップグレードが終了したら、保存されなかったローカル変更を復元しなければならない場合があります。



注意 - 保存されなかったローカル変更を復元しないと、システムがブートされないことがあります。

▼ アップグレード後にローカル変更を復元する方法

1. 次のファイルの内容を確認して、**Solaris suninstall** プログラムによって保存されなかったローカル変更を復元する必要があるかどうかを判断します。

```
/a/var/sadm/system/data/upgrade_cleanup
```

2. 保存されなかったローカル変更があれば、復元します。
3. リブートします。

```
# reboot
```


第 16 章

フラッシュインストール機能

ここでは、フラッシュアーカイブを作成する方法と、フラッシュアーカイブを使用して複数のシステムに Solaris オペレーティング環境をインストールする方法について説明します。

第 17 章	フラッシュ (Flash) アーカイブの作成とインストールの概要と計画について説明します。
第 18 章	フラッシュアーカイブの作成手順を説明します。
第 19 章	フラッシュアーカイブを使用してシステムをインストールする手順を説明します。
第 20 章	<code>flarcreate</code> と <code>flar</code> コマンドの構文およびオプションについて説明します。また、フラッシュアーカイブ情報のキーワードについても説明します。

第 17 章

フラッシュの概要と計画

この章では、フラッシュインストール機能の概要を述べるとともに、使用している環境におけるフラッシュインストールの計画に必要な情報を記載します。

- 107 ページの「フラッシュの概要」
- 108 ページの「フラッシュインストールの計画」

フラッシュの概要

フラッシュインストール機能を使用すると、マスターシステムと呼ばれるシステム上の Solaris オペレーティング環境のアーカイブを作成し、このアーカイブをクローンシステムと呼ばれる多数のシステムに複製できます。フラッシュインストールによるクローンシステムへのインストールは、以下の 3 つのプロセスで行います。

1. マスターシステムへのインストール - システムを 1 台選択し、Solaris の任意のインストール方法で Solaris オペレーティング環境およびその他のソフトウェアをインストールします。
2. フラッシュアーカイブの作成 - フラッシュアーカイブには、マスターシステム上のすべてのファイルのコピーが含まれます。
3. フラッシュアーカイブのクローンシステムへのインストール - システムにフラッシュアーカイブをインストールすると、アーカイブのすべてのファイルがそのシステムにインストールされます。新しくインストールされたシステムのインストール構成は、オリジナルのマスターシステムと同じになります (このため、クローンシステムと呼ばれます)。

注 - フラッシュインストール機能を使用して Solaris オペレーティング環境が稼働しているシステムをアップグレードすることはできません。行えるのは初期インストールだけです。

フラッシュインストールの計画

フラッシュアーカイブの作成とインストールを行う前に、システムに Solaris オペレーティング環境をどのようにインストールするかについていくつか決定しておくべきことがあります。

マスターシステムへのインストールの設計

フラッシュインストールの最初の作業は、各クローンシステムで行いたい構成と同じ構成でマスターシステムにインストールすることです。マスターシステムへの Solaris オペレーティング環境のインストール (部分インストールまたは全インストール) は、Solaris の任意のインストール方法で行うことができます。このインストールの完了後、ソフトウェアの追加または削除、構成ファイルの変更を行います。

マスターシステムとクローンシステムとで、カーネルアーキテクチャを同じにする必要があります。たとえば、アーキテクチャが sun4u であるクローンシステムへインストールするには、同じ sun4u のアーキテクチャを持つマスターシステムから作成されたフラッシュアーカイブしか使用できません。

各クローンシステムで行いたい特別な構成がある場合は、その構成をマスターシステムにインストールする必要があります。マスターシステムへのインストールの設計は、次の要素で決まります。

- クローンシステムにインストールしたいソフトウェア
- マスターシステムとクローンシステムに接続されている周辺装置
- マスターシステムとクローンシステムのアーキテクチャ

マスターシステムのカスタマイズ

Solaris の任意のインストール方法でマスターシステムに Solaris オペレーティング環境をインストールした後、必要に応じてソフトウェアの追加や削除、およびシステム構成情報の変更を行います。

- ソフトウェアの削除 - クローンシステムにインストールする必要がないソフトウェアを削除することができます。マスターシステムにインストールされているソフトウェアのリストを表示するには、Solaris Product Registry を使用します。手順の詳細は、『Solaris のシステム管理 (基本編)』を参照してください。
- ソフトウェアの追加 - Solaris リリースに含まれているソフトウェア、および Solaris オペレーティング環境の一部として提供されないソフトウェアをインストールできます。マスターシステムにインストールされたすべてのソフトウェアはフラッシュアーカイブに含まれ、クローンシステムにインストールされます。
- 構成ファイルの変更 - マスターシステム上の構成ファイルを変更できます。たとえば、`/etc/inet/inetd.conf` ファイルを変更することにより、システムが実行するデーモンを制限できます。加えられた変更はすべてフラッシュアーカイブの一

部として保存され、クローンシステムにインストールされます。

注 - 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したあと、ホスト固有のファイルがいくつか削除され、クローンマシン用として作成し直されます。インストールプログラムは、`sys-unconfig`(1M) コマンドと `sysidtool`(1M) プログラムを使用して、ホスト固有のネットワーク構成ファイルを削除して再作成します。再作成されるファイルは、`/etc/hosts`、`/etc/defaultrouter`、`/etc/defaultdomain` などです。

マスターシステムに存在しない周辺装置のサポート

クローンシステムとは異なる周辺装置を持つマスターシステムにインストールする場合があります。コアシステムサポート (Core) ソフトウェアグループ、エンドユーザーシステムサポート (End User) ソフトウェアグループ、開発者システムサポート (Developer) ソフトウェアグループ、または全体ディストリビューション (Entire) ソフトウェアグループをマスターシステムにインストールした場合、マスターシステムがサポートするのはインストール時にマスターシステムに接続されていた周辺装置だけです。

たとえば、`cg6` フレームバッファを持つマスターシステムに全体ディストリビューション (Entire) ソフトウェアグループをインストールした場合、このインストールでは `cg6` フレームバッファのサポートしか含まれていません。したがって、このシステムから作成したアーカイブは、`cg6` フレームバッファを持つか、フレームバッファを持たないクローンシステムにしかインストールできません。このアーカイブを使用して `Elite3D` フレームバッファを持つクローンシステムにインストールしても、`Elite3D` は使用できません。これは、必要なドライバがインストールされないためです。

マスターシステムに存在しない周辺装置が、クローンシステムには存在するという場合があります。マスターシステムにデバイスが存在しなくても、マスターシステムにこれらの周辺装置のサポートをインストールできます。このマスターシステムから作成するフラッシュアーカイブには、クローンシステム上の周辺装置に対するサポートが含まれます。

マスターシステムとは異なる周辺装置を持つクローンシステムにインストールする予定がある場合には、以下の方法のどれかを使用してマスターシステムにこれらの周辺装置のサポートをインストールすることができます。

- 全体ディストリビューションと OEM サポート (Entire plus OEM) ソフトウェアグループをインストールする: 全体ディストリビューションと OEM サポート (Entire plus OEM) ソフトウェアグループは、利用できる最大のソフトウェアグループであり、Solaris オペレーティング環境のすべてのパッケージが含まれています。マスターシステムに全体ディストリビューションと OEM サポート (Entire plus OEM) ソフトウェアグループをインストールすると、マスターシステムには Solaris リリースに含まれるすべてのドライバが含まれることとなります。全体ディストリビューションと OEM サポート (Entire plus OEM) ソフトウェアグループ

ブをインストールしたマスターシステムから作成されるフラッシュアーカイブは、インストールした Solaris オペレーティング環境のリリースがサポートする周辺装置を持つクローンシステムで機能します。

全体ディストリビューションと OEM サポート (Entire plus OEM) ソフトウェアグループをマスターシステムにインストールすると、ほかの周辺装置構成との互換性を保つことができます。しかし、全体ディストリビューションと OEM サポート (Entire plus OEM) ソフトウェアグループには、1G バイトを超えるディスク容量が必要です。クローンシステムの中には、全体ディストリビューションと OEM サポート (Entire plus OEM) ソフトウェアグループのインストールに必要な容量を確保できないものもあるでしょう。

- パッケージを選択してインストールする: マスターシステムにインストールする際に、マスターシステムとクローンシステムに必要なパッケージを追加インストールできます。特定のパッケージを選択することにより、マスターシステムまたはクローンシステムに存在することが明らかな周辺装置のサポートだけをインストールすることができます。

フラッシュアーカイブの作成の計画

マスターシステムへのインストールの完了後、フラッシュインストールの次の作業として、フラッシュアーカイブを作成します。マスターシステム上のすべてのファイルは、さまざまな識別情報と共にフラッシュアーカイブにコピーされます。フラッシュアーカイブは、マルチユーザーモードまたはシングルユーザーモードでマスターシステムが稼働している間に作成できます。フラッシュアーカイブは、次のメディアのうちの一つからブートした後でも作成できます。

- Solaris 9 DVD
- Solaris 9 SOFTWARE 1 of 2 CD
- Solaris 9 SOFTWARE CD と Solaris 9 LANGUAGES CD のイメージ

アーカイブは、システムができるだけ静的な状態である時に作成してください。

アーカイブ識別情報

フラッシュアーカイブには、クローンシステムにインストールされるマスターシステムからの実際のファイルに加えて、アーカイブ識別情報が含まれます。フラッシュアーカイブの名前はユーザーが指定する必要があります。アーカイブについて指定できる情報には、次のものが含まれます。

- アーカイブの作者
- アーカイブの作成日
- アーカイブを作成したマスターシステムの名前

指定できるアーカイブ識別情報の完全なリストは、126 ページの「identification セクションのキーワード」を参照してください。

アーカイブについての情報を検出するには、`flar` コマンドを使用してください。手順の詳細は、126 ページの「`identification` セクションのキーワード」を参照してください。

フラッシュアーカイブの保存先

フラッシュアーカイブを作成した後、そのアーカイブをマスターシステムのハードディスクまたはテープに保存できます。保存後は、任意のファイルシステムまたは媒体へそのアーカイブをコピーできます。

- ネットワークファイルシステム (NFS) サーバー
- HTTP サーバー
- FTP サーバー
- テープ
- CD、DVD
- フロッピーディスク
- インストールしたいクローンシステムのローカルドライブ

アーカイブの圧縮

フラッシュアーカイブを作成する際は、`compress` (1) ユーティリティを使用して、そのアーカイブが圧縮ファイルとして保存されるように指定することができます。圧縮されたアーカイブはディスク容量が少なく済み、ネットワークを介してアーカイブをインストールする場合の負荷も減ります。

フラッシュアーカイブのインストールの計画

フラッシュインストールの最後の作業は、クローンシステムへのフラッシュアーカイブのインストールです。

フラッシュアーカイブのインストール方法の決定

クローンシステムへのフラッシュアーカイブのインストールには、次のインストール方法を使用できます。

Solaris 9 DVD または Solaris 9 INSTALLATION CD に含まれている Solaris Web Start では、以下の媒体に格納されているフラッシュアーカイブのインストールが可能です。

- ディスク (DVD または CD)
- ネットワークファイルシステム (NFS) サーバー
- HTTP サーバー
- FTP サーバー

- ローカルテープ

Solaris 9 SOFTWARE 1 of 2 CD に含まれている Solaris suninstall プログラムでは、以下の媒体に格納されているフラッシュアーカイブのインストールが可能です。

- HTTP サーバー
- FTP サーバー
- ネットワークファイルシステム (NFS) サーバー
- ローカルファイル
- ローカルテープ
- CD を含むローカルデバイス

カスタム JumpStart インストールプログラムでは、以下の媒体に格納されているフラッシュアーカイブのインストールが可能です。

- ネットワークファイルシステム (NFS) サーバー
- HTTP サーバー
- FTP サーバー
- ローカルテープ
- ローカルデバイス (DVD または CD)
- ローカルファイル

Solaris Live Upgrade を利用すると、次の場所に保存されているフラッシュアーカイブをインストールできます。

- HTTP サーバー
- FTP サーバー
- ネットワークファイルシステム (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)

階層化されたフラッシュアーカイブのインストール

フラッシュ (Flash) インストール機能では、フラッシュアーカイブを階層化する機能を提供します。さまざまな方法でインストールするための、部分的なフラッシュアーカイブを作成することができます。

たとえば、1 つ目に Solaris オペレーティング環境が入ったアーカイブ、2 つ目に Web サーバーの実行に必要なファイルが入ったアーカイブ、3 つ目にネットワークファイルシステム (NFS) サーバー用のファイルが入ったアーカイブを作成します。こうすれば、1 つ目と 2 つ目のアーカイブをインストールして Web サーバーのクローンシステムを作成できます。1 つ目と 3 つ目のアーカイブをほかのシステムにインストールして、ネットワークファイルシステム (NFS) サーバーを作成することもできます。

階層化されたアーカイブを使用すれば、フラッシュインストールの柔軟性を高めながら、フラッシュアーカイブの保存に必要なディスク容量を減らすことができます。階層化されたアーカイブをクローンシステムにインストールする際は、それらのアーカイブの 1 つに Solaris オペレーティング環境が含まれている必要があります。

注 - 階層化されたフラッシュアーカイブを使用してクローンシステムに Solaris オペレーティング環境と追加ソフトウェアを別々にインストールする場合、Solaris パッケージのデータベースには追加ソフトウェアは登録されません。

第 18 章

フラッシュアーカイブの作成

この章では、フラッシュアーカイブを作成する手順を説明します。

- 115 ページの「作業マップ: フラッシュアーカイブを作成する作業マップ」
- 115 ページの「フラッシュアーカイブ作成のための作業」

作業マップ: フラッシュアーカイブを作成する作業マップ

表 18-1 作業マップ: フラッシュアーカイブを作成する作業マップ

作業	説明	参照先
マスターマシンへの、選択した構成でのインストール	ニーズに合った構成を決定し、Solaris の任意のインストール方法を使用してマスターマシンにインストールする。	第 2 章
フラッシュアーカイブの作成	flarcreate コマンドを使用してアーカイブを作成する。	116 ページの「フラッシュアーカイブを作成する方法」

フラッシュアーカイブ作成のための作業

この節では、マスターシステムに Solaris オペレーティング環境をインストールし、その後そのマスターシステムからフラッシュアーカイブを作成する手順を説明します。

▼ マスターシステムにインストールする方法

マスターシステムには、クローンシステムで行いたい構成と同じ構成でインストールします。マスターシステムへの Solaris オペレーティング環境のインストールは、既存の Solaris インストール方法のどの方法でも使用できます。

1. インストールしたいシステム構成を決定します。
2. いずれかの **Solaris** のインストール方法を使用して、マスターシステムに **Solaris** オペレーティング環境をインストールします。インストール方法の選択については、第 3 章を参照してください。
3. マスターシステムを次のいずれかの方法でカスタマイズします。
 - ソフトウェアを削除する
 - ソフトウェアを追加する
 - 構成ファイルを変更する
 - クローンシステム上にある周辺装置のためのサポートを追加する

▼ フラッシュアーカイブを作成する方法

マスターシステムにインストールした後で、クローンシステムへのインストールに使用するフラッシュアーカイブを作成します。

1. マスターシステムをブートし、できるだけ静的な状態で稼働させます。

可能であれば、システムをシングルユーザーモードで実行してください。これが不可能な場合、アーカイブしたいアプリケーションおよび大量のオペレーティングシステムリソースを必要とするアプリケーションを停止します。

フラッシュアーカイブの作成は、マスターシステムがマルチユーザーモードまたはシングルユーザーモードで稼働している間、あるいは次に示すメディアの 1 つからブートする間に行うことができます。

 - Solaris 9 DVD
 - Solaris 9 SOFTWARE 1 of 2 CD
 - Solaris 9 SOFTWARE のイメージ。CD メディアを使用している場合、イメージには必要に応じて Solaris 9 LANGUAGES CD が含まれます。
2. `flarcreate` コマンドを使用して、アーカイブを作成します。

```
# flarcreate -n name options path/filename
```

`name` アーカイブに指定する名前です。指定する `name` は、`content` `_name` キーワードの値になります。

<i>path</i>	アーカイブファイルを保存するディレクトリへのパスです。このパスを指定しない場合、 <code>flarcreate</code> はアーカイブファイルを現在のディレクトリに保存します。
<i>filename</i>	アーカイブファイルの名前です。

コマンド行オプションのリストは、128 ページの「`flarcreate`」を参照してください。

アーカイブの作成が正常に完了すると、`flarcreate` コマンドは終了コード 0 を返します。アーカイブの作成が失敗すると、`flarcreate` コマンドは 0 以外の終了コードを返します。

第 19 章

フラッシュアーカイブのインストール

この章では、フラッシュアーカイブをクローンシステムにインストールする手順を説明します。フラッシュアーカイブは、任意の Solaris のインストール方法でインストールできます。

- 119 ページの「Solaris Live Upgrade によるフラッシュアーカイブのインストール」
- 120 ページの「Solaris Web Start によるフラッシュアーカイブのインストール」
- 121 ページの「Solaris suninstall プログラムによるフラッシュアーカイブのインストール」
- 122 ページの「カスタム JumpStart インストールによるフラッシュアーカイブのインストール」

Solaris Live Upgrade によるフラッシュアーカイブのインストール

Solaris 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方法については、286 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。

Solaris Web Start によるフラッシュアーカイブのインストール

この節では、Solaris 9 DVD または Solaris 9 INSTALLATION CD に含まれている Solaris Web Start を使用して、以下の媒体に格納されているフラッシュアーカイブをインストールする手順を説明します。

- ディスク (DVD または CD)
- ネットワークファイルシステム (NFS) サーバー
- HTTP サーバー
- FTP サーバー
- ローカルテープ

▼ Solaris Web Start 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

1. 第 14 章 に記述されている手順に従って、**Solaris Web Start** インストールを開始します。
2. 「媒体の指定 (**Specify Media**)」画面で、フラッシュアーカイブのロケーションを選択します。
選択された媒体に応じて、Solaris Web Start が続行するためのプロンプトを表示します。
3. プロンプトに従って次の情報を入力します。

選択された媒体	プロンプト
DVD または CD	フラッシュアーカイブを含むディスクを挿入してください。
ネットワークファイルシステム	フラッシュアーカイブを含むネットワークファイルシステムのパスを指定してください。アーカイブのファイル名も指定できます。
HTTP	フラッシュアーカイブにアクセスするための URL とプロキシ情報を指定してください。
FTP	FTP サーバーと、フラッシュアーカイブのパスを指定します。FTP サーバーへのアクセスに必要なユーザーおよびパスワードの情報を指定します。FTP サーバーへのアクセスに必要なプロキシ情報を指定します。

選択された媒体	プロンプト
ローカルテープ	フラッシュアーカイブを含むローカルテープデバイスとテープ上の位置を指定してください。

DVD、CD、または NFS サーバーに格納されたアーカイブのインストールを選択した場合は、「フラッシュアーカイブの選択 (Select Flash Archives)」画面が表示されます。

4. ディスクまたは NFS サーバーに格納されたアーカイブについては、「フラッシュアーカイブの選択 (Select Flash Archives)」画面で、インストールするフラッシュアーカイブを選択します (1 つでも複数でも可)。
5. 「フラッシュアーカイブの一覧 (Flash Archives Summary)」画面で選択されたアーカイブを確認し、「次へ (Next)」をクリックします。
6. 「追加するフラッシュアーカイブ (Additional Flash Archives)」画面で、別のアーカイブを含む媒体を指定して、階層化されたフラッシュアーカイブのインストールを選択することができます。アーカイブをそれ以上インストールしない場合は、「なし - アーカイブの選択が完了しました (None)」を選択し、「次へ (Next)」をクリックしてインストールを続けます。

Solaris suninstall プログラムによる フラッシュアーカイブのインストール

この節では、Solaris 9 SOFTWARE 1 of 2 CD に含まれる Solaris suninstall プログラムを使用して、以下に格納されているフラッシュアーカイブをインストールする手順を説明します。

- HTTP サーバー
- FTP サーバー
- ネットワークファイルシステム (NFS) サーバー
- ローカルファイル
- ローカルテープ
- CD を含むローカルデバイス

▼ Solaris suninstall プログラム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

1. 第 15 章の説明に従って、Solaris suninstall プログラムを起動します。

2. 「フラッシュアーカイブの検索方法 (Flash Archive Retrieval Method)」画面で、フラッシュアーカイブのロケーションを選択します。
 選択された媒体に応じて、Solaris 9 Solaris Web Start インストールが実行するためのプロンプトが表示されます。
3. プロンプトに従って次の情報を入力します。

選択された媒体	プロンプト
HTTP	フラッシュアーカイブへのアクセスに必要な URL とプロキシ情報を指定してください。
FTP	FTP サーバーと、フラッシュアーカイブのパスを指定します。FTP サーバーへのアクセスに必要なユーザーおよびパスワードの情報を指定します。FTP サーバーへのアクセスに必要なプロキシ情報を指定します。
ネットワークファイルシステム	フラッシュアーカイブを含むネットワークファイルシステムのパスを指定してください。アーカイブのファイル名も指定できます。
ローカルファイル	フラッシュアーカイブを含むローカルファイルシステムのパスを指定してください。
ローカルテープ	フラッシュアーカイブを含むローカルテープデバイスとテープ上の位置を指定してください。
ローカルデバイス	フラッシュアーカイブを含むローカルデバイスとパス、ファイルシステムのタイプを指定してください。

4. 「フラッシュアーカイブの選択 (Flash Archive Selection)」画面で、「新規 (New)」を選択すると、階層化されたフラッシュアーカイブのインストールを選択することができます。アーカイブをそれ以上インストールしない場合は、インストールを完了させるために「継続 (Continue)」を選択します。

カスタム JumpStart インストールによる フラッシュアーカイブのインストール

この節では、カスタム JumpStart インストールを使用して、フラッシュアーカイブをインストールする手順を説明します。

- ネットワークファイルシステム (NFS) サーバー
- HTTP サーバー
- FTP サーバー
- ローカルテープ

- ローカルデバイス (DVD または CD)
- ローカルファイル

▼ カスタム JumpStart インストー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

1. インストールサーバーで、カスタム **JumpStart** の `rules` ファイルを作成します。
カスタム **JumpStart** ファイルの作成手順の詳細は、第 23 章を参照してください。
2. インストールサーバーで、カスタム **JumpStart** のプロファイルを作成します。
 - a. `install_type` プロファイルキーワードの値を、`flash_install` と設定します。
 - b. 新しい `archive_location` 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へのパスを追加します。
`archive_location` プロファイルキーワードの詳細は、204 ページの「`archive_location` プロファイルキーワード」を参照してください。
 - c. ファイルシステム構成を指定します。
フラッシュアーカイブの抽出プロセスでは、パーティションの自動配置はサポートされません。
 - d. (省略可能) クローンシステムに階層化されたフラッシュアーカイブをインストールしたい場合は、インストールしたいアーカイブごとに `archive_location` 行を指定してください。

注 – 第 23 章に記載されているカスタム **JumpStart** のプロファイルキーワードのリストの中で、フラッシュアーカイブをインストールする時に有効なキーワードは、以下のもののみです。

- `filesystem - filesystem` プロファイルキーワードに値 `auto` は設定できません。
 - `install_type` (必須)
 - `partitioning - partitioning` プロファイルキーワードに、値 `explicit` または `existing` のみ設定できます。
-

3. インストールサーバーで、フラッシュアーカイブをインストールするクライアントを追加します。
手順の詳細は 88 ページの「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。
4. クローンシステムへのカスタム **JumpStart** インストールを実行します。
手順の詳細は、第 26 章を参照してください。

フラッシュのカスタム JumpStart プログラムファイルの例

次に、カスタム JumpStart インストー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ために利用できるプロファイル例を示します。

次のプロファイル例は、カスタム JumpStart プログラムが HTTP サーバーからフラッシュアーカイブを取り出すことを示しています。

例 19-1 HTTP サーバーからフラッシュアーカイブをインストールする

```
install_type flash_install
archive_location http installserver /flasharchive/solaris9archive
partitioning explicit
filesys c0t1d0s0 4000 /
filesys c0t1d0s1 512 swap
filesys c0t1d0s7 free /export/home
```

次のプロファイル例は、カスタム JumpStart プログラムが NFS サーバーからフラッシュアーカイブを取り出すことを示しています。

例 19-2 NFS サーバーからフラッシュアーカイブをインストールする

```
install_type flash_install
archive_location nfs installserver:/export/solaris/flasharchive/solaris9archive
partitioning explicit
filesys rootdisk.s0 6000 /
filesys rootdisk.s1 512 swap
filesys rootdisk.s7 free /export/home
```

第 20 章

フラッシュに関するリファレンス情報

フラッシュインストール機能では、フラッシュアーカイブの管理および制御のためのツールを提供します。アーカイブを作成後、アーカイブ情報を抽出する、1つのアーカイブを複数のセクションに分割する、複数のアーカイブセクションを結合するなどの作業が行えます。

- 125 ページの「フラッシュアーカイブのセクション」
- 126 ページの「フラッシュのキーワード」
- 128 ページの「フラッシュのコマンド」

フラッシュアーカイブのセクション

フラッシュアーカイブには、少なくとも3つのセクションが含まれます。これらのセクションは、アーカイブ識別情報と、クローンシステムにインストールされるマスターシステムからコピーされた実際のファイルから構成されています。

1. **cookie** セクション - フラッシュアーカイブの先頭には、ファイルをフラッシュアーカイブとして識別する `cookie` が含まれます。アーカイブが有効であるためには、この `cookie` が存在する必要があります。
2. **identification** セクション - 2つ目のセクションには、アーカイブについての識別情報を示す値が指定されたキーワードが含まれます。
3. ユーザー定義のセクション - **identification** セクションの後には、ユーザーがセクションを定義して挿入できます。フラッシュアーカイブは、ユーザーが挿入したセクションは処理しません。ユーザー定義のセクションは、行単位で作成し、復帰改行 (ASCII 0x0a) 文字で終了させる必要があります。個々の行の長さに制限はありません。ユーザー定義のセクションにバイナリデータを含める場合は、base64 またはそれに類似したアルゴリズムを使用してコード化する必要があります。

ユーザー定義のセクションの名前は「X」で始める必要がありますが、その後には改行、等号、空文字、フォワードスラッシュ (/) 以外の任意の文字を続けることができます。たとえば、「X-department」はユーザー定義のセクション名として有

効です。

4. **archive** セクション - archive セクションには、マスターシステムから保存されたファイルが含まれます。

フラッシュのキーワード

キーワードと値は1つの等号で区切られ、1行に1キーワードを指定します。個々の行の長さに制限はありません。キーワードは、大文字と小文字を区別する必要があります。

一般的なキーワード

フラッシュアーカイブの各セクションの初めと終わりは、`section_begin` と `section_end` キーワードで定義されます。`section_begin` と `section_end` キーワードの値を以下の表に示します。

表 20-1 `section_begin` と `section_end` キーワードの値

アーカイブのセクション	<code>section_begin</code> と <code>section_end</code> キーワードの値
cookie セクション	<code>cookie</code>
identification セクション	<code>identification</code>
ユーザー定義のセクション	<code>section_name</code>
archive セクション	<code>archive</code>

identification セクションのキーワード

この節では、`identification` セクションで使用されるキーワードと、これらに定義できる値について説明します。

次の表は、アーカイブについて記述するキーワードを説明しています。

表 20-2 identification セクションのキーワード

キーワード	値の定義
content_name (必須)	<p>フラッシュアーカイブの展開ユーティリティは、content_name キーワードの値を使用してアーカイブを識別します。この値は、256 文字以内で指定する必要があります。</p> <p>content_name キーワードの値は、アーカイブの選択と抽出プロセスの間ユーザーに提示されることがあるため、この値にアーカイブの機能と目的を記述すると便利です。</p>
creation_date	<p>creation_date キーワードの値は、アーカイブが作成された日時を示す、テキスト形式のタイムスタンプです。値は、YYYYMMDDhhmmss 形式にしてください。たとえば、20000131221409 は、2000 年 1 月 31 日、22 時 14 分 09 秒を示します。作成日を指定しない場合、デフォルトの日付がグリニッジ標準時 (GMT) で設定されます。</p>
creation_master	<p>creation_master キーワードの値は、アーカイブの作成に使用したマスターシステムの名前です。creation_master に値を指定しない場合、flarcreate には uname -n が出力するシステム名が設定されます。</p>
content_type	<p>content_type キーワードの値は、アーカイブのカテゴリを指定するために定義します。フラッシュアーカイブの展開ユーティリティは、展開中に content_type キーワードの値を表示します。</p>
content_description	<p>content_description キーワードの値は、アーカイブの内容を記述するために定義します。このキーワードの値の長さに制限はありません。</p>
content_author	<p>content_author キーワードの値は、アーカイブの作成者を示すために定義します。作成者のフルネームと電子メールアドレスを含めることをお勧めします。</p>
content_architectures	<p>content_architectures キーワードの値は、アーカイブがサポートするカーネルアーキテクチャをコンマで区切って指定したリストです。フラッシュアーカイブを作成すると、このアーカイブによって content_architectures キーワードの値が生成されます。</p> <p>アーカイブにこのキーワードが含まれる場合、フラッシュアーカイブ展開ユーティリティは、アーカイブがサポートするアーキテクチャのリストに照らしてクローンシステムのカーネルアーキテクチャの検証を行います。アーカイブがクローンシステムのカーネルアーキテクチャをサポートしていない場合、展開は失敗します。このキーワードが存在しない場合、展開ユーティリティはクローンシステムのアーキテクチャの検証を行いません。</p>
creation_node	<p>uname -n コマンドが返す値。作成されたフラッシュアーカイブのルートディレクトリが / でないと、flarcreate は nodename (4) ファイルの内容を使用します。</p>
creation_hardware_class	<p>uname -m コマンドが返す値。作成されたフラッシュアーカイブのルートディレクトリが / でないと、flarcreate は UNKNOWN という文字列を挿入します。</p>

表 20-2 identification セクションのキーワード (続き)

キーワード	値の定義
creation_platform	uname - i コマンドが返す値。作成されたフラッシュアーカイブのルートディレクトリが / でないと、flarcreate は UNKNOWN という文字列を挿入します。
creation_processor	uname - p コマンドが返す値。作成されたフラッシュアーカイブのルートディレクトリが / でないと、flarcreate は UNKNOWN という文字列を挿入します。
creation_release	uname - r コマンドが返す値。作成されたフラッシュアーカイブのルートディレクトリが / でないと、flarcreate は <i>root</i> _directory/var/sadm/system/admin/INST_RELEASE の内容を使用しようとします。flarcreate は、このファイルを読み取ることができないと、UNKNOWN という文字列を挿入します。
creation_os_name	uname - s コマンドが返す値。作成されたフラッシュアーカイブのルートディレクトリが / でないと、flarcreate は <i>root</i> _directory/var/sadm/system/admin/INST_RELEASE の内容を使用しようとします。flarcreate は、このファイルを読み取ることができないと、UNKNOWN という文字列を挿入します。
creation_os_version	uname - v コマンドが返す値。作成されたフラッシュアーカイブのルートディレクトリが / でないと、flarcreate は UNKNOWN という文字列を挿入します。

フラッシュアーカイブで定義されるキーワードに加えて、ユーザー自身でもキーワードを定義できます。フラッシュアーカイブはユーザー定義のキーワードを無視しますが、identification セクションを処理しユーザー定義のキーワードを使用するスクリプトまたはプログラムを提供できます。ユーザー定義のキーワードの名前は「X」で始まる必要がありますが、改行、等号、空文字以外の任意の文字を含むことができます。たとえば、X-department はユーザー定義のキーワードとして有効です。

フラッシュのコマンド

フラッシュのコマンドは、フラッシュアーカイブの作成と管理に使用します。

flarcreate

flarcreate コマンドは、マスターシステムからフラッシュアーカイブを作成するために使用します。このコマンドは、マスターシステムがマルチユーザーモードまたはシングルユーザーモードで稼働している時に使用できます。また、flarcreate は、マスターシステムが Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD からブート

している時、あるいは Solaris 9 SOFTWARE および Solaris 9 LANGUAGES CD の ネットイメージからブートしている時にも使用できます。フラッシュアーカイブを作成する時は、マスターシステムはできるだけ静的な状態にしておく必要があります。このコマンドの構文は次のとおりです。

```
flarcreate -n name [-R root] [-S] [-H] [-c] [-x exclude] [-t [-p posn ]
[-b blocksize]] [-i date] [-m [- u section [-d dir]] [-f file_list] [-F] [-U
key=val] master] [-a author] [-e descr:-E descr_file] [-T type] path/filename
```

このコマンド行で、*path* は、アーカイブファイルを保存するディレクトリです。*filename* は、アーカイブファイルの名前です。パスを指定しない場合、*flarcreate* はアーカイブファイルを現在のディレクトリに保存します。

表 20-3 flarcreate のコマンド行オプション

オプション	説明
必須オプション	
-n <i>name</i>	このオプションの値は、アーカイブの名前です。指定する <i>name</i> は、 <i>content_name</i> キーワードの値になります。
圧縮用のオプション	
-c	<i>compress(1)</i> を使用してアーカイブを圧縮します。
ディレクトリとサイズを指定するオプション	
-R <i>root</i>	<i>root</i> のファイルシステムツリーからアーカイブを作成します。このオプションを指定しない場合、 <i>flarcreate</i> は / ファイルシステムからアーカイブを作成します。
-S	アーカイブにサイズ情報を含めません。
-H	ハッシュ識別子を生成しません。
ファイルやディレクトリを除外するオプション	
-x <i>exclude</i>	ファイルまたはディレクトリ <i>exclude</i> をアーカイブの対象外にします。ファイルシステムに -R <i>root</i> を指定すると、ディレクトリ <i>exclude</i> へのパスは <i>root</i> からの相対パスとみなされます。
ユーザー定義のセクションで使用されるオプション	
-u <i>section</i>	ユーザー定義のセクションを含めます。ユーザー定義のセクションを複数含めるには、セクション名を空白で区切ったリストにします。
-d <i>dir</i>	<i>dir</i> から、-u で指定したセクションを取り出します。
テープアーカイブで使用されるオプション	
-t	テープデバイス上にアーカイブを作成します。 <i>filename</i> 引数は、テープデバイスの名前です。

表 20-3 flarcreate のコマンド行オプション (続き)

オプション	説明
-p <i>posn</i>	-t オプションとの併用でのみ使用できます。flarcreate がアーカイブを格納するための、テープデバイス上の位置を指定します。このオプションを指定しない場合、flarcreate はテープの現在の位置にアーカイブを配置します。
-b <i>blocksize</i>	flarcreate がアーカイブの作成時に使用するブロックサイズを指定します。ブロックサイズを指定しない場合、flarcreate はデフォルトのブロックサイズ 64k を使用します。
ファイルを指定するためのオプション	
-f <i>file_list</i>	<i>file_list</i> に示されるファイルをアーカイブに追加します。 <i>file_list</i> ファイルでは、1 ファイルを 1 行で表す必要があります。各ファイルへのパスは、代替ルートディレクトリからの相対パスまたは絶対パスです。 <i>file_list</i> の値として「-」を指定すると、ファイルのリストとして stdin の出力が使用されます。「-」を指定すると、アーカイブのサイズは計算されません。
-F	<i>file_list</i> に示されるファイルだけを使ってアーカイブを作成します。
アーカイブ識別のためのオプション	
-U <i>key=val</i>	identification セクションに、ユーザー定義のキーワードと値を含めます。
-i <i>date</i>	<i>date</i> は、creation_date キーワードの値として使用されます。 <i>date</i> を指定しない場合、flarcreate は現在のシステム日時を使用します。
-m <i>master</i>	<i>master</i> は、creation_master キーワードのアーカイブを作成したマスターシステムの名前として使用されます。 <i>master</i> を指定しない場合、flarcreate は <code>uname -n</code> が出力するシステム名を使用します。
-e <i>descr</i>	<i>descr</i> は、content_description キーワードの値として使用されます。-E オプションを使用する場合はこのオプションを使用できません。
-E <i>descr_file</i>	ファイル <i>descr_file</i> から content_description キーワードの値を取得します。-e オプションを使用する場合はこのオプションを使用できません。
-a <i>author</i>	<i>author</i> は、identification セクション内の content_author キーワードの値として使用されます。作成者を指定しない場合、flarcreate は identification セクションの content_author キーワードを含めません。

表 20-3 flarcreate のコマンド行オプション (続き)

オプション	説明
-T <i>type</i>	<i>type</i> は、 <code>content_type</code> キーワードの値として使用されます。タイプを指定しない場合、 <code>flarcreate</code> は <code>content_type</code> キーワードを含めません。

flar

`flar` コマンドは、アーカイブの管理に使用します。このコマンドで行うことができる作業は以下のとおりです。

- 131 ページの「アーカイブからの情報の抽出」
- 131 ページの「アーカイブの分割」
- 132 ページの「アーカイブの結合」

アーカイブからの情報の抽出

作成済みのアーカイブについての情報を取得するには、`-i` または `info` オプションを指定して `flar` コマンドを使用します。このコマンドの構文は次のとおりです。

```
flar -i:info [- l] [-k keyword] [- t [-p posn] [- b blocksize]] filename
```

表 20-4 `flar -i` (`flar info`) のコマンド行オプション

オプション	説明
-k <i>keyword</i>	キーワード <i>keyword</i> の値のみを返します。
-l	アーカイブセクション内のすべてのファイルをリスト表示します。

アーカイブの分割

フラッシュアーカイブを複数のセクションに分割するには、`-s` または `split` オプションを指定して `flar` コマンドを使用します。`flar` コマンドは、現在のディレクトリまたは指定されたディレクトリ内に各セクションを個々のファイルに分けてコピーします。ファイルには、セクション名からとった名前が付けられます。たとえば、`cookie` セクションは、`cookie` と名付けられたファイルに保存されます。`flar` コマンドでは、1つのセクションだけを保存するように指定できます。このコマンドの構文は次のとおりです。

```
flar -s:split[- d dir] [-u section] [-f archive] [-S section] [-t [-p posn] [-b blocksize]] filename
```

表 20-5 flar -s (flar split) のコマンド行オプション

オプション	説明
-d <i>dir</i>	コピーするセクションを、現在のディレクトリからではなく <i>dir</i> から取得します。
-u <i>section</i>	このオプションを使用しない場合、flar は現在のディレクトリ内のすべてのセクションをコピーします。このオプションを使用した場合、flar は <i>cookie</i> セクション、 <i>identification</i> セクション、 <i>archive</i> セクション、および <i>section</i> セクションをコピーします。1 つのセクション名を指定することも、あるいは空白で区切って複数のセクション名を指定することも可能です。
-f <i>archive</i>	アーカイブセクションを、 <i>archive</i> という名前のファイルに入れるのではなく、 <i>archive</i> というディレクトリに抽出します。
-S <i>section</i>	アーカイブから <i>section</i> と名付けられたセクションだけをコピーします。

アーカイブの結合

個々のセクションからフラッシュアーカイブを作成するには、-c または *combine* オプションを指定して、flar コマンドを使用します。各セクションは、セクション名を名前に持つ個々のファイル内にあると見なされます。少なくとも、*cookie* (*cookie* セクション)、*identification* (*identification* セクション)、および *archive* (*archive* セクション) の 3 つのファイルが存在する必要があります。*archive* がディレクトリである場合、flar コマンドは、結合されたアーカイブに含める前に、*cpio* を使用してこれをアーカイブします。*identification* セクションにアーカイブの圧縮が指定されている場合、flar は新しく結合されたアーカイブのコンテンツを圧縮します。

```
flar -c:combine [-d dir] [-u section] [-t [-p posn] [-b blocksize]] filename
```

注 - どのセクションについても検証は何も行われません。特に、*identification* セクション内のどのフィールドについても、検証や更新は行われません。

表 20-6 flar -c (flar combine) のコマンド行オプション

オプション	説明
-d <i>dir</i>	結合するセクションを、現在のディレクトリからではなく <i>dir</i> から取得します。

表 20-6 flar -c (flar combine) のコマンド行オプション (続き)

オプション	説明
<code>-u section</code>	このオプションを指定しない場合、flar は現在のディレクトリ内のすべてのセクションを結合します。このオプションを使用した場合、flar は <code>cookie</code> セクション、 <code>identification</code> セクション、 <code>archive</code> セクション、および <code>section</code> セクションのみを結合します。1 つのセクション名を指定することも、あるいは空白で区切って複数のセクション名を指定することも可能です。

第 21 章

カスタム JumpStart インストール

以下の章では、カスタム JumpStart インストールのための準備や実行手順について説明します。

第 22 章	カスタム JumpStart インストールの紹介と概要を示します。
第 23 章	各サイトでカスタム JumpStart インストールを使用して Solaris 9 ソフトウェアをインストールするために必要な準備の手順について説明します。
第 24 章	カスタム JumpStart インストールツールで実行するスクリプトを作成するときに使用できるオプション機能について説明します。
第 25 章	独自のカスタムルールとカスタムプロンプトを作成するための情報と手順を説明します。
第 26 章	カスタム JumpStart インストールを実行する方法を説明します。これらの手順は、Solaris 9 ソフトウェアをインストールしようとするシステム上で実行してください。
第 27 章	カスタム JumpStart を使用して Solaris ソフトウェアの設定とインストールを行う例を示します。
第 28 章	rules ファイル、プロファイル、begin スクリプト、および finish スクリプトで使用できるキーワードと値を示します。

第 22 章

カスタム JumpStart の概要

この章では、カスタム JumpStart のインストール処理について紹介し、全体の概要を示します。

- 137 ページの「カスタム JumpStart とは」
- 138 ページの「JumpStart プログラムが Solaris ソフトウェアをインストールする方法」

カスタム JumpStart とは

カスタム JumpStart は、あらかじめ作成されたプロファイルを使用して、自動的に複数のシステムをインストールまたはアップグレードできるコマンド行インタフェースです。プロファイルには、どのようにソフトウェアをインストールするかを定義します。さらに、インストール前とインストール後に実行する作業を、シェルスクリプトを使用して指定することができます。システムのインストールまたはアップグレードにどのプロファイルとスクリプトを使用するかをユーザーが選択すると、それらの選択に基づいて処理が行われます。また、`sysidcfg` ファイルを使用して構成情報を指定することにより、カスタム JumpStart インストールを完全に自動化することも可能です。

以下に、1 例を挙げてカスタム JumpStart の処理を説明します。この事例では、システムは次のパラメータを指定して設定する必要があるとします。

- 100 台の新しいシステムに Solaris をインストールする。
- 70 台のシステムはエンジニアリンググループが所有し、各システムにはスタンドアロンシステムとして「開発者システムサポート」ソフトウェアグループをインストールする必要がある。
- 残りの 30 台のシステムはマーケティンググループが所有し、各システムにはスタンドアロンシステムとして「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をインストールする必要がある。

まず、システム管理者はシステムグループごとに rules ファイルとプロファイルを作成する必要があります。rules ファイルは、Solaris ソフトウェアをインストールするシステムグループごと(または1つのシステム)のルールが入ったテキストファイルです。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、各グループをプロファイルにリンクします。

プロファイルは、グループ内の各システムに Solaris ソフトウェアがどのようにインストールされるかを定義するテキストファイルです。rules ファイルとプロファイルは、JumpStart ディレクトリに置かれている必要があります。

この例の場合、システム管理者は2つの異なるルールで rules ファイルを作成します。1つはエンジニアリンググループ用のルールで、もう1つはマーケティンググループ用のルールです。ルールごとに、エンジニアリンググループとマーケティンググループを区別するために、システムのネットワーク番号が使用されます。

各ルールには、適切なプロファイルへのリンクも含まれています。たとえば、エンジニアリンググループ用のルールでは、エンジニアリンググループ用に作成した eng_profile というプロファイルへのリンクを追加します。マーケティンググループ用のルールでは、マーケティンググループ用に作成した market_profile というプロファイルへのリンクを追加します。

rules ファイルとプロファイルは、フロッピーディスクまたはサーバー上に保存できます。

- プロファイルフロッピーディスクは、カスタム JumpStart インストールを、ネットワークに接続されていないスタンドアロンシステムで実行するときが必要です。
- プロファイルサーバーは、カスタム JumpStart インストールを、サーバーにアクセスできるネットワークに接続されたシステムで実行するときを使用します。

rules ファイルとプロファイルを作成した後、check スクリプトを使用して、これらのファイルの妥当性を検査する必要があります。check スクリプトが正常に動作する場合、rules.ok ファイルが作成されます。rules.ok は rules ファイルの生成バージョンであり、JumpStart プログラムによって Solaris ソフトウェアのインストールに使用されます。

JumpStart プログラムが Solaris ソフトウェアをインストールする方法

rules ファイルとプロファイルの検証が終わると、カスタム JumpStart インストールを開始できます。JumpStart プログラムは、rules.ok ファイルを読み取ります。続いて、Solaris ソフトウェアがインストールされるシステムに一致するシステム属性を持つ最初のルールを検索します。一致するルールが見つかったら、JumpStart プログラムはそのルール内に指定されているプロファイルを使用してシステムに Solaris ソフトウェアをインストールします。

図 22-1 は、ネットワークに接続されていないスタンドアロン型のシステムでカスタム JumpStart インストールがどのように行われるかを示しています。この例では、システム管理者は Pete のシステムでカスタム JumpStart インストールを開始しています。JumpStart プログラムは、システムのコピーディスクドライブに挿入されているコピーディスク上の rules ファイルにアクセスし、rule 2 をシステムに対応付けます。rule 2 には、JumpStart プログラムが Pete のプロファイルを使用して Solaris ソフトウェアをインストールするように指定されています。JumpStart プログラムは、Pete のプロファイルを読み取り、このプロファイル内に指定されている指示に基づいて Solaris ソフトウェアをインストールします。

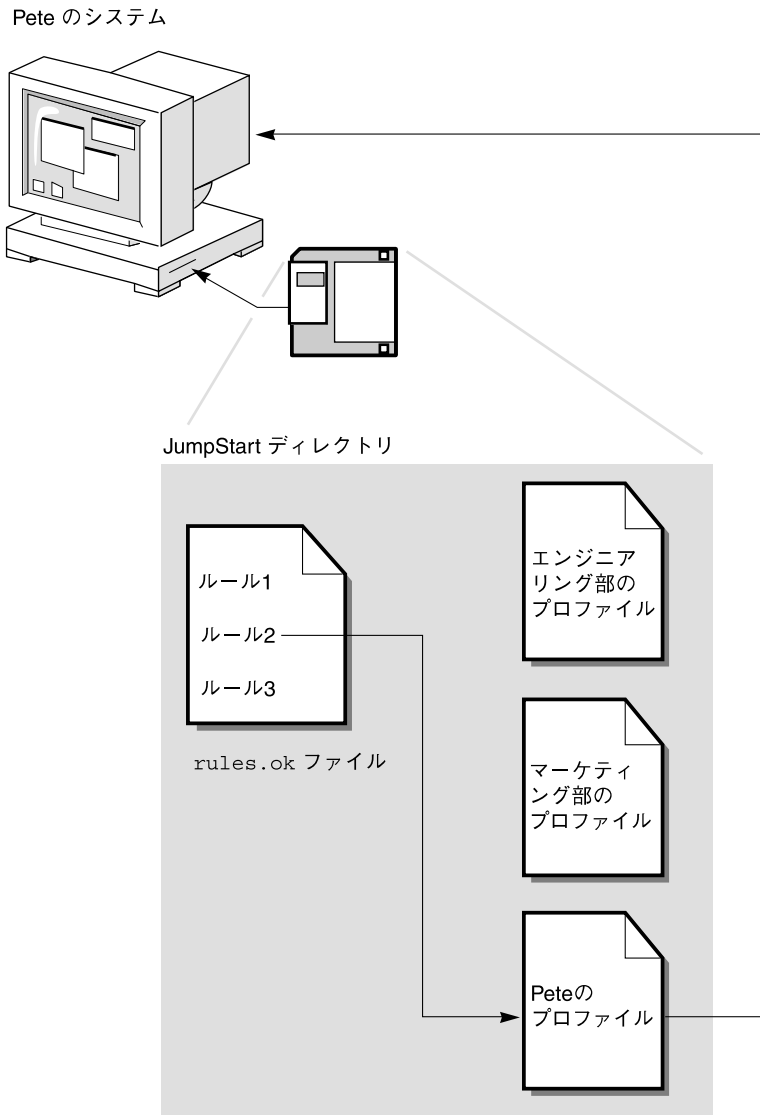


図 22-1 カスタム JumpStart インストールの動作: ネットワークに接続されていないシステムの例

図 22-2 は、ネットワーク上の複数のシステムが存在する場合にカスタム JumpStart インストールがどのように行われるかを示しています。すでに個別のプロファイルが設定され、1 台のサーバーにまとめて保存されています。システム管理者は、エンジニアリング部のシステムの中の 1 台でカスタム JumpStart インストールを開始します。JumpStart プログラムは、サーバー上の JumpStart/ ディレクトリに存在する rules ファイルにアクセスします。JumpStart プログラムは、エンジニアリング部のシ

システムを rule 1 に対応付けます。rule 1 には、JumpStart プログラムがエンジニアリング部のプロファイルを使用してSolaris ソフトウェアをインストールするように指定されています。JumpStart はエンジニアリング部のプロファイルを読み取り、このプロファイル内に指定されている指示に基づいて Solaris ソフトウェアをインストールします。

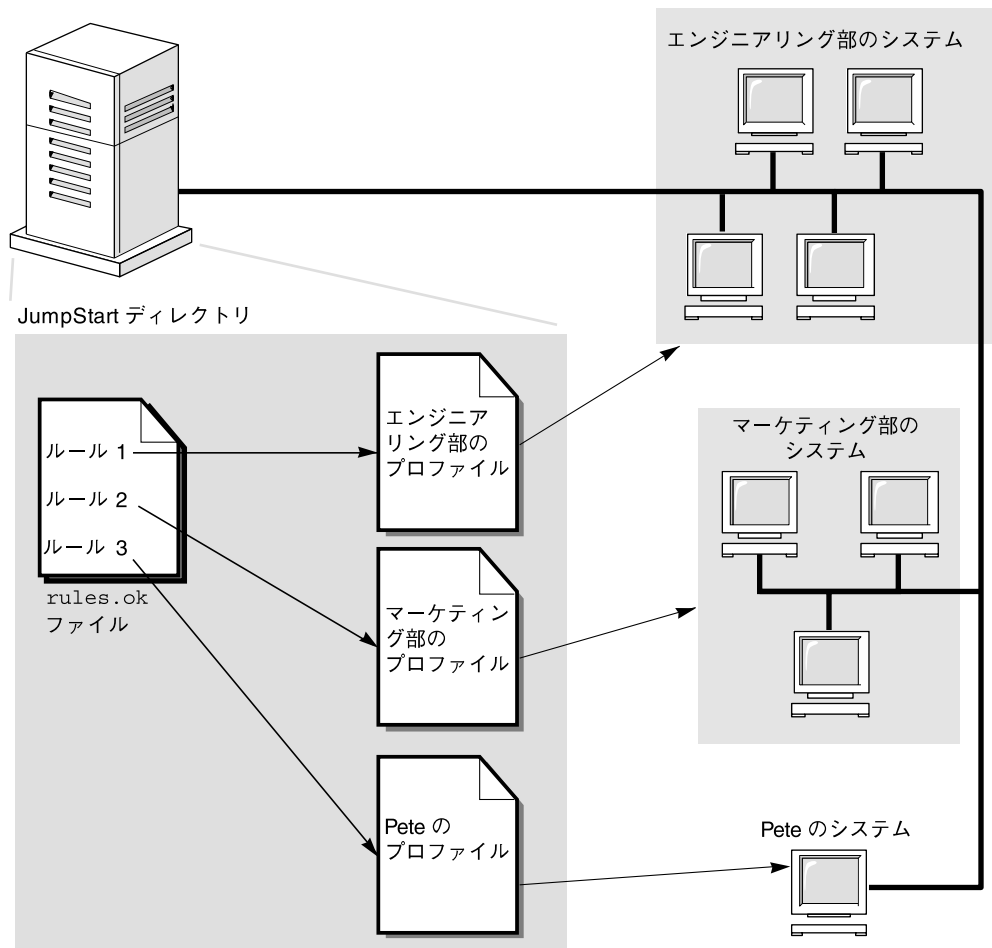


図 22-2 カスタム JumpStart インストールの動作: ネットワークに接続されているシステムの例

図 22-3 は、JumpStart プログラムがカスタム JumpStart ファイルを検索する順番を示しています。

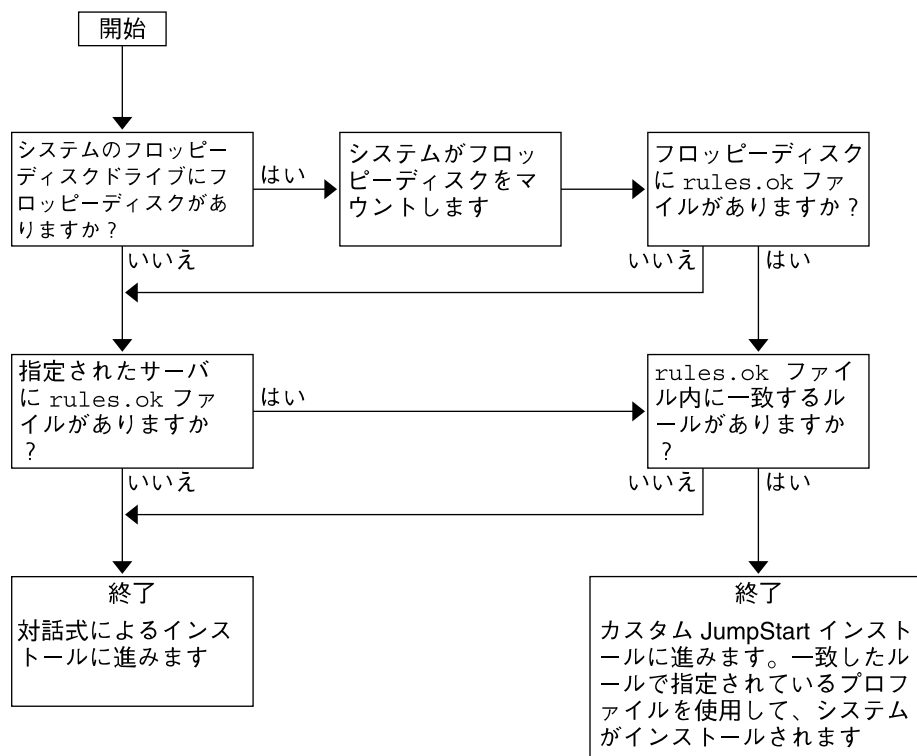


図 22-3 カスタム JumpStart インストール実行時の流れ

第 23 章

カスタム JumpStart インストールの準備

この章では、カスタム JumpStart インストールを使用して、各サイトのシステムに Solaris 9 ソフトウェアをインストールするために必要な準備の手順について説明します。

- 144 ページの「作業マップ: カスタム JumpStart インストールの準備」
- 145 ページの「ネットワーク上のシステム用のプロファイルサーバーの作成」
- 149 ページの「スタンドアロンシステム用のプロファイルフロッピーディスクの作成」
- 152 ページの「rules ファイルの作成」
- 155 ページの「プロファイルの作成」
- 159 ページの「プロファイルのテスト」
- 163 ページの「rules ファイルの妥当性を検査する」

作業マップ: カスタム JumpStart インストールの準備

表 23-1 作業マップ: カスタム JumpStart インストールの準備

作業	説明	参照先
旧バージョンの Solaris ソフトウェアがシステム上にインストールされている場合は、システムをアップグレードする方法を決定する	システムに以前の Solaris リリースがインストールされている場合は、システムをどのようにアップグレードするかを決める必要があります。システムアップグレードの前と後に実施すべき作業を明確にしておいてください。計画を立てることによって、プロファイル、begin スクリプト、finish スクリプトなどの作成が容易になります。	第 8 章
JumpStart ディレクトリを作成する	<p>サーバーの場合</p> <p>ネットワーク接続されたシステムに対してカスタマイズ JumpStart インストールを実行する場合は、プロファイルサーバーを作成する必要があります。プロファイルサーバーには、カスタム JumpStart ファイル用の JumpStart ディレクトリが入っています。</p> <p>フロッピーディスクの場合</p> <p>ネットワークに接続されていないシステムに対して JumpStart インストールを実行する場合は、プロファイルフロッピーディスクを作成する必要があります。プロファイルフロッピーディスクには、カスタム JumpStart ファイルが入っています。</p>	<p>145 ページの「ネットワーク上のシステム用のプロファイルサーバーの作成」</p> <p>149 ページの「スタンドアロンシステム用のプロファイルフロッピーディスクの作成」</p>
ルールを rules ファイルに追加する	各システムグループまたは単一のシステムをどのようにインストールするか決定した後で、インストールする各グループに対してルールを作成します。各ルールは、1 つ以上のシステム属性に基づいてグループを識別し、それぞれのグループを 1 つのプロファイルにリンクします。	152 ページの「rules ファイルの作成」

表 23-1 作業マップ: カスタム JumpStart インストールの準備 (続き)

作業	説明	参照先
ルールごとにプロファイルを作成する	プロファイルは、どのように Solaris ソフトウェアをシステムにインストールするか (たとえば、どのソフトウェアグループをインストールするか) を定義するテキストファイルです。すべてのルールはプロファイルを指定して、ルールが一致したときにシステムがどのようにインストールされるかを定義します。通常は、ルールごとに異なるプロファイルを作成しますが、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。	155 ページの「プロファイルの作成」
(省略可能) プロファイル进行测试する	プロファイルの作成後、pfinstall (1M) コマンドを使用すれば、プロファイルを使用してシステムをインストールまたはアップグレードする前に、プロファイル进行测试できます。	159 ページの「プロファイルのテスト」
rules ファイルの妥当性を検査する	rules.ok ファイルは、rules ファイルから生成されたファイルで、JumpStart プログラムがプロファイルを使用してインストールするシステムを一致させるために使用されます。rules ファイルの妥当性を検査するには、check スクリプトを使用する必要があります。	163 ページの「rules ファイルの妥当性を検査する」

ネットワーク上のシステム用のプロファイルサーバーの作成

ネットワーク上のシステム用にカスタム JumpStart インストールを設定する際には、サーバー上に JumpStart ディレクトリと呼ばれるディレクトリを作成する必要があります。JumpStart ディレクトリには、重要なカスタム JumpStart ファイル (rules ファイル、rules.ok ファイル、プロファイルなど) がすべて置かれます。JumpStart ディレクトリは、プロファイルサーバーのルート (/) ディレクトリに作成する必要があります。

JumpStart ディレクトリを持つサーバーは、「プロファイルサーバー」と呼びます。プロファイルサーバーは、インストールサーバーまたはブートサーバーと同じにすることも、あるいは異なるサーバーとして独立させることもできます。

注 - プロファイルサーバーの作成後、システムがそのサーバーにアクセスできるように設定する必要があります。詳細は、147 ページの「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する方法」を参照してください。

▼ JumpStart ディレクトリをサーバー上に作成する方法

注 - この手順では、システムがボリュームマネージャを実行していると仮定していません。ボリュームマネージャを使用せずにメディアを管理する方法については、Solaris 9 System Administrator Collection に含まれている『Solaris のシステム管理 (基本編)』を参照してください。

1. **JumpStart** ディレクトリを作成するサーバーに、スーパーユーザーとしてログインします。

2. サーバーに **JumpStart** ディレクトリを作成します。

```
# mkdir -m 755 jumpstart_dir_path
```

jumpstart_dir_path は、**JumpStart** ディレクトリの絶対パスです。

たとえば、次のコマンドは、ルートファイルシステムに **jumpstart** というディレクトリを作成し、アクセス権を 755 に設定します。

```
# mkdir -m 755 /jumpstart
```

3. `/etc/dfs/dfstab` ファイルを編集します。次のエントリを追加してください。

```
share -F nfs -o ro,anon=0 jumpstart_dir_path
```

たとえば、次のエントリは `/jumpstart` ディレクトリを共有します。

```
share -F nfs -o ro,anon=0 /jumpstart
```

4. `shareall` と入力して **Enter** を押します。

5. サンプルのカスタム **JumpStart** ファイルを **JumpStart** ディレクトリにコピーするかどうかを決定します。

- コピーしない場合は、手順 8 へ進みます。
- コピーする場合は、次の表を利用して次に行う作業を決定します。

例のコピー元	手順
プラットフォームに対応した Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD	サーバーの CD-ROM ドライブに、Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD を挿入します。 ボリュームマネージャが自動的に CD をマウントします。
ローカルディスク上で使用されているプラットフォーム用の Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD	Solaris 9 DVD イメージまたは Solaris 9 SOFTWARE 1 of 2 イメージが配置されている場所にディレクトリを移動します。 コマンド例を示します。 <code>cd /export/install</code>

6. サンプルのカスタム **JumpStart** ファイルを、プロファイルサーバーの **JumpStart** ディレクトリにコピーします。

```
# cp -r media_path/Solaris_9/Misc/jumpstart_sample/* jumpstart_dir_path
```

<i>media_path</i>	ディスク上の CD、DVD、またはイメージのパス。
<i>jumpstart_dir_path</i>	カスタム JumpStart ファイルの例を配置する、プロファイルサーバー上のパス。

たとえば、次のコマンドは、`jumpstart_sample` ディレクトリをプロファイルサーバー上の `/jumpstart` ディレクトリにコピーします。

```
cp -r /cdrom/cdrom0/s0/Solaris_9/Misc/jumpstart_sample/* /jumpstart
```

7. サンプル **JumpStart** ファイルを更新し、それらのファイルがサイトの環境内で動作するようにします。
8. **JumpStart** ディレクトリの所有者が `root` で、そのアクセス権が **755** に設定されていることを確認します。
9. ネットワーク上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設定します。
詳細は、147 ページの「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する方法」を参照してください。

▼ 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する方法

プロファイルサーバーを作成する際に、システムがカスタム **JumpStart** インストール中にプロファイルサーバーにアクセスできるようにする必要があります。このためには、次の方法があります。

- `add_install_client` コマンド – ネットワークインストーラでシステムを追加するたびに、`add_install_client` コマンドの `-c` オプションを使用します。手順の詳細は 88 ページの「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。
- `boot` コマンド – システムのブート時にプロファイルサーバー上の `JumpStart` ディレクトリの場所を指定します。カスタム `JumpStart` 構成ファイルは、圧縮して 1 つのファイルにする必要があります。続いて、NFS サーバー、HTTP サーバー、またはシステムがローカルにアクセスできるメディアに圧縮されたその構成ファイルを保存します。詳細は、172 ページの「圧縮された構成ファイルの作成」を参照してください。

システムをブートしてカスタム `JumpStart` インストールを開始する場合は、この圧縮ファイルの場所を指定してください。手順の詳細は、184 ページの「カスタム `JumpStart` プログラムを使用してインストールまたはアップグレードする方法」の手順 5 を参照してください。
- `/etc/bootparams` ファイル – `/etc/bootparams` ファイルでワイルドカードを使用します。`etc/bootparams` ファイルにワイルドカードを追加するには、以下の手順に従ってください。

注 – 以下の手順は、`JumpStart` ディレクトリをフロッピーディスクに保存する場合と、システムのブート時にプロファイルサーバーの場所を指定する場合は不要です。

この手順は、ネットワークインストール情報が `/etc/bootparams` ファイルに格納されている場合だけ有効です。ネットワークインストール情報は、次の場所に格納することもできます。

- ネームサービスデータベース – ネットワークインストール情報がネームサービスの `bootparams` データベースに格納されている場合は、手順 3 に示すエントリを `bootparams` データベースに追加する必要があります。
 - DHCP サーバー – ネットワークインストール情報が DHCP サーバーに格納されている場合は、`boot` コマンドを使用してカスタム `JumpStart` が DHCP サーバーを使用するように指定する必要があります。手順の詳細は、184 ページの「カスタム `JumpStart` プログラムを使用してインストールまたはアップグレードする方法」の手順 5 を参照してください。
1. インストールサーバーまたはブートサーバーにスーパーユーザーとしてログインします。
 2. テキストエディタを使用して `/etc/bootparams` を開きます。
 3. 次のエントリを追加します。

* `install_config=server:jumpstart_dir_path`

* すべてのシステムにアクセスできるように指定するワイルドカード文字

server JumpStart ディレクトリがあるプロファイルサーバーのホスト名

jumpstart_dir_path JumpStart ディレクトリの絶対パス

たとえば、次のエントリはすべてのシステムが、*sherlock* というサーバーにある */jumpstart* ディレクトリにアクセスできるようにします。

```
* install_config=sherlock:/jumpstart
```



注意 - この手順を使用した場合、インストールクライアントを起動したときに次のエラーメッセージが表示されることがあります。

WARNING: getfile: RPC failed: error 5: (RPC Timed out).

このエラーメッセージの詳細は、347 ページの「ネットワークからのブート時のエラーメッセージ」を参照してください。

これで、すべてのシステムはプロファイルサーバーにアクセスできるようになりました。

スタンドアロンシステム用のプロファイルフロッピーディスクの作成

JumpStart ディレクトリを含むフロッピーディスクは、プロファイルフロッピーディスクと呼ばれます。ネットワークに接続されていないシステムには、プロファイルサーバーへのアクセス権はありません。このため、システムがネットワークに接続されていない場合はフロッピーディスク上に JumpStart ディレクトリを作成する必要があります。プロファイルフロッピーディスクを作成するシステムには、フロッピーディスクドライブが必要です。

JumpStart ディレクトリには、重要なカスタム JumpStart ファイル (*rules* ファイル、*rules.ok* ファイル、プロファイルなど) がすべて入っています。JumpStart ディレクトリは、プロファイルフロッピーディスクのルート (*/*) ディレクトリに作成する必要があります。

▼ プロファイルフロッピーディスクを作成する方法

注 – この手順では、システムがボリュームマネージャ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を使用せずにフロッピーディスク、CD、およびDVDを管理する方法の詳細は、Solaris 9 System Administrator Collection に含まれている『Solaris のシステム管理 (基本編)』を参照してください。

1. フロッピーディスクドライブを持つシステムにスーパーユーザーとしてログインします。
2. 空のフロッピーディスクまたは上書き可能なディスクをフロッピーディスクドライブに挿入します。
3. フロッピーディスクをマウントします。

```
# volcheck
```

4. フロッピーディスクに **UNIX** ファイルシステム (**UFS**) が含まれているかどうかを確認します。
システム上のファイル `/etc/mnttab` の内容を調べ、次のようなエントリがないか確認します。

```
/vol/dev/diskette0/scrap /floppy/scrap ufs suid,rw,largefiles,dev=1740008 927147040
```

- このようなエントリが存在する場合は、手順7へ進みます。
- エントリが存在しない場合は、次の手順へ進みます。

5. フロッピーディスクをフォーマットします。



注意 – この手順によって、フロッピーディスク上のすべてのデータは上書きされます。

```
# fdformat -U
```

6. フロッピーディスクに **UFS** ファイルシステムを作成します。

```
# newfs /vol/dev/aliases/floppy0
```

7. サンプルのカスタム **JumpStart** ファイルを **JumpStart** ディレクトリにコピーするかどうかを決定します。
 - コピーしない場合は、手順10へ進みます。
 - コピーする場合は、次の表を利用して次に行う作業を決定します。

例のコピー元	手順
Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD (SPARC)	サーバーの CD-ROM ドライブに Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD を挿入します。 ボリュームマネージャが自動的に CD をマウントします。
ローカルディスク上の Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD のイメージ	Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD イメージの場所にディレクトリを移動させます。コマンド例を示します。 <code>cd /export/install</code>

8. サンプルのカスタム **JumpStart** ファイルを、プロファイルフロッピーディスクのルートディレクトリ (**JumpStart** ディレクトリ) にコピーします。

```
# cp -r media_path/Solaris_9/Misc/jumpstart_sample/* jumpstart_dir_path
```

<i>media_path</i>	ディスク上の CD、DVD、またはイメージのパス。
<i>jumpstart_dir_path</i>	カスタム JumpStart のサンプルファイルを置くプロファイルフロッピーディスクのパス。

注 - すべてのカスタム JumpStart インストールファイルは、フロッピーディスクのルートディレクトリに置かれている必要があります。

たとえば、次のコマンドは、Solaris 9 SOFTWARE 1 of 2 CD の `jumpstart_sample` の内容を、プロファイルフロッピーディスク `scrap` のルート (/) ディレクトリにコピーします。

```
cp -r /cdrom/sol_9_sparc/s0/Solaris_9/Misc/jumpstart_sample/* /floppy/scrap
```

9. プロファイルフロッピーディスク上の **JumpStart** ファイルの例を更新して、サイトの環境内で動作するようにします。
10. **JumpStart** ディレクトリの所有者が `root` で、そのアクセス権が **755** に設定されていることを確認します。
11. フロッピーディスクを取り出します。

```
# eject floppy
```

これで、プロファイルフロッピーディスクの作成が完了しました。rules ファイルを更新して、プロファイルフロッピーディスクにプロファイルを作成して、カスタム JumpStart インストールを実行できます。処理を続けるには、152 ページの「rules ファイルの作成」へ進んでください。

rules ファイルの作成

rules ファイルは、Solaris オペレーティング環境のインストール先となる各システムグループのルールを含むテキストファイルです。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、各グループをプロファイルにリンクします。プロファイルは、グループ内の各システムに Solaris ソフトウェアをどのようにインストールするかを示したテキストファイルです。たとえば、次のルールは JumpStart プログラムが basic_prof プロファイル内の情報を使用し、sun4u プラットフォームグループを持つすべてのシステムに対してインストールを実行することを示します。

```
karch sun4u - basic_prof -
```

rules ファイルを使用して、カスタム JumpStart インストールに必要な rules.ok ファイルを作成します。

注 - 149 ページの「スタンドアロンシステム用のプロファイルフロッピーディスクの作成」または 145 ページの「ネットワーク上のシステム用のプロファイルサーバーの作成」の手順にしたがって JumpStart ディレクトリを設定した場合は、JumpStart ディレクトリに rules ファイルのサンプルがあります。rules ファイルのサンプルには、説明といくつかのルール例があります。サンプルの rules ファイルを利用する場合は、使用しないルール例は必ずコメントにしておいてください。

rules ファイルの構文

rules ファイルは、次のように作成する必要があります。

- ファイルには rules という名前を割り当てる。
- ファイルには1つ以上のルールを含める。

rules ファイルには、次の要素を任意に含めることができます。

- コメント
行の # 記号の後に含まれるテキストは、JumpStart ではコメントとして扱われます。行が # 記号で始まる場合は、行全体がコメントとして扱われます。
- 1行以上の空白行
- 1つまたは複数の複数行ルール
1つのルールを新しい行に続けるには、Return キーを押す直前に、バックスラッシュ文字 (\) を含めます。

▼ rules ファイルを作成する方法

1. テキストエディタを使用して `rules` というテキストファイルを作成するか、あるいは作成済みの **JumpStart** ディレクトリからサンプル `rules` ファイルを開きます。
2. **Solaris** ソフトウェアのインストール先となるシステムグループごとに **1** つのルールを `rules` ファイルに追加します。
`rules` ファイルのキーワードと値の一覧は、197 ページの「ルールキーワードと値の説明」を参照してください。
`rules` ファイル内のルールは、次の構文に従っている必要があります。

```
[!]rule_keyword rule_value [&& [!]rule_keyword rule_value] ... begin profile finish
```

表 23-2 ルールの構文要素

要素	説明
!	ルールキーワードの前で使用し、否定を示す記号
<i>rule_keyword</i>	これは、ホスト名 (<code>hostname</code>) やメモリーサイズ (<code>memsize</code>) などの一般的なシステム属性を記述する事前定義された字句単位または語。 <i>rule_keyword</i> は、同じ属性を持つシステムを 1 つのプロファイルに対応させるためにルール値と共に使用される。ルールキーワードの一覧は、197 ページの「ルールキーワードと値の説明」を参照してください。
<i>rule_value</i>	対応するルールキーワードに特定のシステム属性を与える値。ルール値の一覧は、197 ページの「ルールキーワードと値の説明」を参照
&&	ルールキーワードとルール値のペアを同じルールで結合する (論理積をとる) ときに使用する記号。カスタム JumpStart インストール時に、システムがルール内のすべてのペアに一致しなければ、ルールの一致は成立しない
<i>begin</i>	インストール開始前に実行される Bourne シェルスクリプト名。 <i>begin</i> スクリプトがない場合、このフィールドにマイナス記号 (-) を指定する必要がある。 <i>begin</i> スクリプトはすべて、JumpStart ディレクトリになければならない。 <i>begin</i> スクリプトの作成方法の詳細は、165 ページの「 <i>begin</i> スクリプトの作成」を参照
<i>profile</i>	テキストファイル名。システムがルールに一致したとき Solaris ソフトウェアがシステムにどのようにインストールされるかを定義している。プロファイル内の情報は、プロファイルキーワードと、それらに対応するプロファイル値から構成される。すべてのプロファイルは JumpStart ディレクトリになければならない。 注 - プロファイルフィールドについては、別の使用方法もあります。詳細は、175 ページの「サイト固有のインストールプログラムの使用」と 166 ページの「 <i>begin</i> スクリプトによる動的プロファイルの作成」を参照してください。

表 23-2 ルールの構文要素 (続き)

要素	説明
<i>finish</i>	<p>インストール終了後に実行される Bourne シェルスクリプト名。finish スクリプトがない場合、このフィールドにマイナス記号 (-) を指定する必要がある。finish スクリプトはすべて、JumpStart ディレクトリになければならない。</p> <p>finish スクリプトの作成方法の詳細は、167 ページの「finish スクリプトの作成」を参照</p>

各ルールには、少なくとも次の要素を含める必要があります。

- キーワード、値、および対応するプロファイル
- *begin* および *finish* フィールド内のマイナス記号 (-) (*begin* および *finish* スクリプトが指定されない場合)

3. rules ファイルを **JumpStart** ディレクトリに保存します。
4. root が rules ファイルを所有していて、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。

rules ファイルの例

次のサンプルファイルは、rules ファイル内のルールをいくつか示しています。各行には、ルールキーワードとそのキーワードに有効な値があります。JumpStart プログラムは、rules ファイルを上から下へ走査します。

ルールキーワードと値を既知のシステムに対応付けることができた場合、JumpStart プログラムは profile フィールドに示されているプロファイルによって指定されている Solaris ソフトウェアをインストールします。

例 23-1 rule ファイル

```
# rule keywords and rule values      begin script      profile      finish script
# -----
hostname eng-11                      -              basic_prof   -
network 192.43.34.0 && !model \
'SUNW,SPARCstation-20'2              -              net_prof     -
model SUNW,SPARCstation-LX3         -              lx_prof      complete
network 193.144.2.0 && karch sun4u  setup4          4u_prof      done
memsize 64-128 && arch sparc5        -              prog_prof    -
any6 -                               -              generic_prof -
```

1. このルールは、システムのホスト名が eng-1 の場合に一致します。basic_prof プロファイルは、ルールに一致するシステムに Solaris ソフトウェアをインストールするために使用されます。
2. このルールは、システムがサブネットワーク 192.43.34.0 にあって、SPARCstation™ 20 (SUNW, SPARCstation-20) ではない場合に一致します。net_prof プロファイルは、このルールに一致するシステムに Solaris ソフトウェアをインストールする

例 23-1 rule ファイル (続き)

ために使用されます。このルールは、152 ページの「rules ファイルの構文」に説明されているルールラップの例も示しています。

3. このルールは、システムが SPARCstation LX である場合に一致します。lx_prof プロファイルと complete 終了スクリプトは、このルールに一致するシステムに Solaris ソフトウェアをインストールするために使用されます。
4. このルールは、システムがサブネット 193.144.2.0 にあって、sun4U のシステムである場合に一致します。setup 開始スクリプト、4u_prof プロファイル、および done 終了スクリプトは、このルールに一致するシステムに Solaris ソフトウェアをインストールするために使用されます。
5. このルールは、システムのメモリーが 64M バイトから 128M バイトのときに一致します。prog_prof プロファイルは、このルールに一致するシステムに Solaris ソフトウェアをインストールするために使用されます。
6. このルールは、上記のルールに一致しなかったすべてのシステムに一致します。generic_prof プロファイルは、このルールに一致するシステムに Solaris ソフトウェアをインストールするために使用されます。any を使用する場合は、この値を常に rules ファイルの最後のファイルとして使用する必要があります。

プロファイルの作成

プロファイルは、システムに Solaris ソフトウェアをインストールする方法を示したテキストファイルです。プロファイルには、インストール要素 (インストールするソフトウェアグループなど) を指定します。各ルールには、システムのインストール方法を示すプロファイルを指定します。プロファイルは、ルールごとに異なるものを作成しても、複数のルールに同じものを使用しても構いません。

プロファイルは、1 つまたは複数のプロファイルキーワードとそれらの値から構成されます。各プロファイルキーワードは、JumpStart プログラムがどのようにしてシステムに Solaris ソフトウェアをインストールするかを制御するコマンドです。たとえば、次のプロファイルキーワードと値は、JumpStart プログラムがシステムをサーバーとしてインクルードすることを指定しています。

```
system_type server
```

注 - 145 ページの「ネットワーク上のシステム用のプロファイルサーバーの作成」または 149 ページの「スタンドアロンシステム用のプロファイルフロッピーディスクの作成」の手順を使用して JumpStart ディレクトリを作成した場合、プロファイルのサンプルが JumpStart ディレクトリにあります。

プロファイルの構文

プロファイルには次の要素を含める必要があります。

- 最初のエントリが `install_type` プロファイルキーワードであること
- キーワードは1行に1つだけであること
- アップグレードされるシステムが、アップグレードできるルートファイルシステムを複数持っている場合は、`root_device` キーワード

プロファイルには以下の要素を含めることができます。

- コメント
行の # 記号の後に含まれるテキストは、JumpStart ではコメントとして扱われ
ず。行が # 記号で始まる場合は、行全体がコメントとして扱われます。
- 1行以上の空白行

▼ プロファイルを作成する方法

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、わかりやすい名前を付けま
す。または、作成した **JumpStart** ディレクトリ内のサンプルプロファイルを開きま
す。

注 – プロファイルの名前は、システムに 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 (例: `basic
_install`、`eng_profile`、`user_profile`)。

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。
プロファイルのキーワードと値の一覧は、202 ページの「プロファイルキーワードと
値」を参照してください。

注 – プロファイルキーワードとプロファイル値には、大文字と小文字の区別がありま
す。

3. **JumpStart** ディレクトリにプロファイルを保存します。
4. プロファイルの所有権が `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。
5. (省略可能) プロファイルをテストします。
プロファイルのテストに関する情報については、159 ページの「プロファイルのテス
ト」を参照してください。

プロファイルの例

次のサンプルプロファイルは、さまざまなプロファイルキーワードとプロファイル値を使用して、Solaris ソフトウェアをシステムにどのようにインストールするかを指定する方法を示しています。202 ページの「プロファイルキーワードと値」には、プロファイルキーワードと値の説明を示してあります。

例 23-2 リモートファイルシステムのマウントとパッケージの追加および削除

```
# profile keywords      profile values
# -----
install_type           initial_install1
system_type            standalone2
partitioning           default3
filesys                any 512 swap # specify size of /swap
cluster                SUNWCprog4
package               SUNWman delete5
cluster                SUNWCacc
```

1. `install_type` キーワードは、すべてのプロファイルに必要です。
2. `system_type` キーワードは、システムをスタンドアロンシステムとしてインストールするように定義します。
3. ファイルシステムスライスは、インストールするソフトウェアごとに `default` 値を使用して決定されます。`swap` は 512M バイトに設定され、値 `any` に基づいてすべてのディスクにインストールされます。
4. 「開発者システムサポートソフトウェアグループ」(SUNWCprog) がシステムにインストールされます。
5. 標準のニュアルページはネットワークのファイルサーバー (`s_ref`) からマウントされるため、マニュアルページパッケージはシステムにインストールされません。ただし、System Accounting ユーティリティが入ったパッケージはシステムにインストールされます。

例 23-3 ファイルシステムのインストール場所の指定

```
# profile keywords      profile values
# -----
install_type           initial_install
system_type            standalone
partitioning           explicit1
filesys                c0t0d0s0 auto /
filesys                c0t3d0s1 auto swap
filesys                any auto usr
cluster                SUNWCall2
```

1. `partitioning` の値が `explicit` であるため、ファイルシステムスライスは、`filesys` キーワードによって指定します。ルート (/) のサイズは、値 `auto` により選択したソフトウェア (値 `auto`) に基づいて自動的に設定されて `c0t0d0s0` にインストールされます。`swap` も同様に、必要なサイズに設定された上で `c0t3d0s1` 上にインストールされます。`usr` は、値 `any` により選択したソフト

例 23-3 ファイルシステムのインストール場所の指定 (続き)

ウェアに基づきインストールプログラムによってインストール先が決定されます。

2. Entire Solaris Software Group (SUNWCa11) がシステムにインストールされます。

例 23-4 アップグレード用ディスク容量の再配置

```
# profile keywords      profile values
# -----
install_type           upgrade1
root_device            c0t3d0s22
backup_media           remote_filesystem timber:/export/scratch3
layout_constraint      c0t3d0s2 changeable 1004
layout_constraint      c0t3d0s4 changeable
layout_constraint      c0t3d0s5 movable
package                SUNWbcp delete5
package                SUNWxwman add6
cluster                SUNWCacc add
locale                 de7
```

1. このプロファイルは、ディスク容量を再配置することによってシステムをアップグレードします。この例では、システム上のファイルシステムのいくつかにアップグレード用の容量が十分でないため、ディスク領域を再配置する必要があります。
2. c0t3d0s2 のルートファイルシステムがアップグレードされます。
3. リモートシステム timber が、ディスク容量の再配置中のデータのバックアップに使用されます。
4. layout_constraint キーワードは、アップグレードのためにディスク領域の再配置を行う時に自動レイアウトが以下のことを実施できることを示します。
 - スライス 2 と 4 を変更する。これらのスライスはほかの場所へ移動が可能であり、サイズも変更できます。
 - スライス 5 を移動する。スライスをほかの場所へ移動することはできますが、そのサイズを変更することはできません。
5. バイナリ互換パッケージ (SUNWbcp) は、アップグレード後、システムにインストールされません。
6. このコードは、X Window System のマニュアルページと、System Accounting ユーティリティがまだシステムにインストールされていない場合に、インストールされるようにするものです。すでにシステム上にあるすべてのパッケージが自動的にアップグレードされます。
7. ドイツ語ローカライズパッケージがシステムにインストールされます。

プロファイルのテスト

プロファイルを作成した後で、`pfinstall(1M)` コマンドを使用してプロファイルをテストします。プロファイルのテストは、実際にプロファイルを使用してシステムのインストールまたはアップグレードを行う前に行なってください。

`pfinstall` が生成するインストール出力を調べることによって、意図するとおりにプロファイルが動作するかを簡単に調べることができます。たとえば、実際にシステムでアップグレードを行う前に、そのシステムが Solaris の新しいリリースにアップグレードするための十分なディスク容量を持っているかどうかをプロファイルで調べることができます。

`pfinstall` を利用すると、以下の要素に対してプロファイルをテストできます。

- `pfinstall` を実行しているシステムのディスク構成
- その他のディスク構成。ディスクの構造 (ディスクのバイト/セクタ、フラグ、スライスなど) を示すディスク構成ファイルを使用します。ディスク構成ファイルの作成方法については、173 ページの「ディスク構成ファイルの作成」を参照してください。

注 - ディスク構成ファイルを使用して、システムのアップグレードに使用するプロファイルをテストすることはできません。代わりに、システムの実際のディスク構成およびシステムに現在インストールされているソフトウェアに対して、プロファイルをテストする必要があります。

▼ 一時的な Solaris 9 環境を作成してプロファイルをテストする方法

特定の Solaris リリースでプロファイルを正常かつ正確にテストするには、同じリリースの Solaris 環境内にあるプロファイルをテストする必要があります。たとえば、Solaris 9 初期インストールプロファイルをテストする場合は、Solaris 9 を実行しているシステムに対して `pfinstall` コマンドを実行する必要があります。

以下のどちらかの状況でプロファイルをテストする場合は、一時的なインストール環境を作成する必要があります。

- 旧バージョンの Solaris ソフトウェアを使用しているシステム上で Solaris 9 のアップグレードプロファイルをテストする場合。
- Solaris 9 の初期インストールプロファイルをテストするための Solaris 9 システムをまだインストールしていない場合。

1. 次に示すメディアのいずれか 1 つのイメージからシステムをブートします。

- Solaris 9 DVD
- Solaris 9 SOFTWARE 1 of 2 CD

注 – アップグレードプロファイルをテストしたい場合は、アップグレードするシステムをブートしてください。

2. システムを識別するための質問に答えます。
3. インストールプログラムを終了します。
 - Solaris 9 DVD を使用している場合は、次のプロンプトに対して ! を入力します。

```
Solaris Web Start will assist you in installing software for Solaris.
<Press ENTER to continue> {"!" exits}
```
 - Solaris 9 SOFTWARE 1 of 2 CD を使用している場合は、Solaris 9 オペレーティング環境をインストールするプログラムとして Solaris suninstall プログラムを選択します。表示される最初の画面で「Exit」を選択します。
4. このシェルから pfinstall コマンドを実行します。pfinstall コマンドの使用方法については、160 ページの「プロファイル进行测试する方法」の手順 7 を参照してください。

▼ プロファイル进行测试する方法

1. プロファイル进行测试するためのシステムを決定します。
 アップグレードプロファイル进行测试する場合、アップグレードする実際のシステム上のプロファイル进行测试してください。
2. 次の表に基づいて、次に行うことを決定します。

状態	手順
初期インストールプロファイル进行测试する (Solaris 9 ソフトウェアが動作しているシステムがある場合)	そのシステムでスーパーユーザーになり、手順 5 へ進みます。
アップグレードプロファイル进行测试する (初期インストールプロファイル进行测试するための Solaris 9 が動作しているシステムがない場合)	一時的な Solaris 9 環境を作成し、プロファイル进行测试します。詳細は、159 ページの「一時的な Solaris 9 環境を作成してプロファイル进行测试する方法」を参照してください。続いて、手順 3 へ進みます。

3. 一時的なマウントポイントを作成します。

```
# mkdir /tmp/mnt
```


4. テストしたいプロファイル (1 つ以上) が入ったディレクトリをマウントします。

マウントするメディア	コマンド
リモート NFS ファイルシステムをマウントする (ネットワーク上のシステムの場合)	<code>mount -F nfs server_name:path /tmp/mnt</code>
UFS フォーマットのフロッピーディスクをマウントする	<code>mount -F ufs /dev/diskette /tmp/mnt</code>

5. システムのメモリーサイズを指定してプロファイル进行测试するには、SYS_MEMSIZE に使用するメモリーサイズを **M** バイトで設定します。

```
# SYS_MEMSIZE=memory_size
# export SYS_MEMSIZE
```

6. 手順 4 でディレクトリを

- マウントした場合は、/tmp/mnt にディレクトリを変更します。

```
# cd /tmp/mnt
```

- マウントしなかった場合は、プロファイルのある場所にディレクトリを変更します。通常は、JumpStart ディレクトリです。

```
# cd jumpstart_dir_path
```

7. pfinstall (1M) コマンドによってプロファイル进行测试します。

```
# /usr/sbin/install.d/pfinstall -D:-d disk_config_file [-c path] profile
```



注意 `--d` または `-D` オプションを指定してください。いずれかを指定しないと `pfinstall` は指定されたプロファイルを使用して Solaris 9 ソフトウェアをインストールします。その結果、システム上のデータがすべて上書きされます。

-D

`pfinstall` は、現在のシステムのディスク構成を使用してプロファイル进行测试します。アップグレードプロファイル进行测试する場合は、`-D` オプションを使用する必要があります。

-d <i>disk_config_file</i>	<p>pfinstall は、ディスク構成ファイル <i>disk_config_file</i> を使用してプロファイル进行测试します。pfinstall が動作しているディレクトリに <i>disk_config_file</i> が置かれていない場合は、パスを指定する必要があります。</p> <p>ディスク構成ファイルの作成方法については、173 ページの「ディスク構成ファイルの作成」を参照してください。</p> <p>注 - アップグレードプロファイル (install_type upgrade) で -d <i>disk_config_file</i> オプションを使用することはできません。常に、システムのディスク構成に対してアップグレードプロファイル进行测试する必要があります (つまり、-D オプションを使用する必要があります)。</p>
-c <i>path</i>	<p>Solaris 9 ソフトウェアイメージのパス。このオプションは、たとえば、システムでボリュームマネージャを使用して Solaris 9 SOFTWARE 1 of 2 CD を各自のプラットフォームにマウントする場合に使用します。</p> <p>注 - プラットフォーム対応の Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD イメージからブートした場合は、-c オプションは不要です。DVD または CD イメージは、ブート処理の一環として /cdrom 上にマウントされます。</p>
<i>profile</i>	<p>テストするプロファイル名。<i>profile</i> が pfinstall が実行されているディレクトリにない場合は、パスを指定する必要があります。</p>

プロファイルテストの例

次の例は、pfinstall を使用して basic_prof というプロファイル进行测试する方法を示しています。このテストは、Solaris 9 ソフトウェアがインストールされているシステム上のディスク構成に対して行われます。basic_prof プロファイルは、/jumpstart ディレクトリにあります。ボリュームマネージャを使用しているので、Solaris 9 SOFTWARE 1 of 2 CD イメージへのパスが指定されています。

例 23-5 Solaris 9 システムによるプロファイルテスト

```
# cd /jumpstart
# /usr/sbin/install.d/pfinstall -D -c /cdrom/pathname basic_prof
```

次の例は、pfinstall を使用して Solaris 9 システム上の basic_prof というプロファイル进行测试する方法を示しています。535_test ディスク構成ファイルがテストされ、システムメモリーが 64M バイトあるか確認されます。また、この例で pfinstall は、/export/install ディレクトリにある Solaris 9 SOFTWARE 1 of 2 CD のイメージを使用します。

例 23-6 ディスク構成ファイルによるプロファイルテスト

```
# SYS_MEMSIZE=64
# export SYS_MEMSIZE
# /usr/sbin/install.d/pfinstall -d 535_test -c /export/install basic_prof
```

rules ファイルの妥当性を検査する

rules ファイルとプロファイルは、check スクリプトを実行して、これらのファイルが正しく設定されていることを確認してからでないと使用できません。すべてのルールとプロファイルが有効な場合は、rules.ok ファイルが作成されます。このファイルは、カスタム JumpStart インストールソフトウェアがシステムをプロファイルと照合するために必要です。

表 23-3 に check スクリプトの動作を示します。

表 23-3 check スクリプトを使用した場合の動作

手順	説明
1	rules ファイルの構文検査を行う。 check は、ルールキーワードが正当かどうか、および各ルールに対して <i>begin</i> フィールド、 <i>class</i> フィールド、および <i>finish</i> フィールドが指定されているかどうかを確認する(<i>begin</i> および <i>finish</i> フィールドはファイル名でなくマイナス記号 (-) であることもある)。
2	rules ファイルにエラーがなければ、ルールに指定された各プロファイルの構文検査が行われる。
3	エラーがなければ、check スクリプトは rules ファイルから rules.ok ファイルを作成する。コメントと空白行をすべて削除し、すべてのルールを保持し、終わりに次のコメント行を追加する。 # version=2 checksum=num

▼ rules ファイルの妥当性を検査する方法

1. **check** スクリプトが **JumpStart** ディレクトリにあることを確認します。ディスクの構造 (たとえば、ディスクのバイトまたはセクター、フラグ、スライスなど) を表すディスク構成ファイルを使用して、他のディスク構成に対して。

注 - check スクリプトは、Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD の Solaris_9/Misc/jumpstart_sample ディレクトリにあります。

2. **JumpStart** ディレクトリへ移動します。
3. **check** スクリプトを実行して rules ファイルの妥当性を検査します。

```
$ ./check [-p path -r file_name]
```

- `-p path` 使用しているシステムの check スクリプトではなく Solaris 9 ソフトウェアイメージの check スクリプトを使用して `rules` を検証します。 `path` は、ローカルディスク、マウントされた Solaris 9 DVD、または Solaris 9 SOFTWARE 1 of 2 CD 上のイメージです。
- システムが以前のバージョンの Solaris を実行している場合、このオプションを使用して、最新バージョンの check スクリプトを実行します。
- `-r file_name` 名前が `rules` 以外の `rules` ファイル名を指定します。このオプションを使用すると、`rules` ファイルに組み込む前にルールの妥当性を検査できます。

check スクリプトを実行すると、`rules` ファイルの有効性と各プロファイルの有効性の検査結果が表示されます。エラーが検出されないと、`The custom JumpStart configuration is ok` と表示されます。

4. `root` が `rules.ok` ファイルを所有していて、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。

`rules` ファイルの検証が終わると、第 24 章でオプションのカスタム JumpStart 機能について参照してください。カスタム JumpStart インストールの実行については、第 26 章を参照してください。

第 24 章

カスタム JumpStart オプション機能の使用

この章では、カスタム JumpStart インストールのオプション機能について説明します。

- 165 ページの「begin スクリプトの作成」
- 167 ページの「finish スクリプトの作成」
- 172 ページの「圧縮された構成ファイルの作成」
- 173 ページの「ディスク構成ファイルの作成」
- 175 ページの「サイト固有のインストールプログラムの使用」

begin スクリプトの作成

「begin スクリプト」とは、ユーザーが定義する Bourne シェルスクリプトで、rules ファイル内で指定し、Solaris ソフトウェアがシステムにインストールされる前に作業を実行します。begin スクリプトは、カスタム JumpStart インストールで Solaris をインストールするときのみ使用できます。

begin スクリプトは、次のような場合に使用してください。

- 動的プロファイルを作成する
- アップグレード前にファイルのバックアップをとる

begin スクリプトに関する注意事項

- スクリプト内には、初期インストールまたはアップグレードインストール時に /a に対して行われるファイルシステムのマウントを妨げるような記述をしないでください。JumpStart プロファイルが /a にファイルシステムをマウントできない場合は、エラーが発生してインストールは失敗します。
- begin スクリプトの出力は、/var/sadm/begin.log に保存されます。

- root が begin スクリプトを所有していて、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。
- begin スクリプト内では、カスタム JumpStart 環境変数を使用できます。環境変数の一覧は、226 ページの「カスタム JumpStart の環境変数」を参照してください。
- begin スクリプトは、JumpStart ディレクトリに保存してください。

begin スクリプトによる動的プロファイルの作成

「動的プロファイル」とは、カスタム JumpStart インストール時に begin スクリプトが動的に作成するプロファイルです。動的プロファイルは、特定のシステムを 1 つのプロファイルに対応させる rules ファイルを設定できない場合に必要となります。たとえば、異なるハードウェア構成要素を持つ同一のシステムモジュール (異なるフレームバッファを持つシステムなど) には、動的プロファイルを使用する必要があります。

動的プロファイルを使用するルールの設定手順は次のとおりです。

- プロファイルの代わりに、プロファイルフィールドに等号 (=) を指定します。
- インストール対象のシステムに依存する動的プロファイルを作成する begin スクリプトを begin フィールドに指定します。

システムが、プロファイルフィールドに等号 (=) があるルールと一致すると、begin スクリプトは、システムに Solaris ソフトウェアをインストールするのに使用する動的プロファイルを作成します。

次に、毎回同じ動的プロファイルを作成する begin スクリプトの例を示します。もちろん、ルールの評価に応じて異なる動的プロファイルを生成する begin スクリプトも作成できます。

例 24-1 動的プロファイルを作成する begin スクリプト

```
#!/bin/sh
echo "install_type          initial_install"> ${SI_PROFILE}
echo "system_type          standalone">> ${SI_PROFILE}
echo "partitioning         default">> ${SI_PROFILE}
echo "cluster              SUNWCprog">> ${SI_PROFILE}
echo "package              SUNWman      delete">> ${SI_PROFILE}
echo "package              SUNWolman   delete">> ${SI_PROFILE}
echo "package              SUNWxwman   delete">> ${SI_PROFILE}
```

たとえば、begin スクリプトは、デフォルトで /tmp/install.input に設定されている動的プロファイル名に SI_PROFILE 環境変数を使用する必要があります。

注 – begin スクリプトを動的プロファイルの作成に使用する場合は、スクリプトにエラーがないことを確認してください。動的プロファイルは、begin スクリプトが実行されてから作成されます。このため、check スクリプトで動的プロファイルの妥当性は検査されません。

finish スクリプトの作成

「finish スクリプト」とは、ユーザーが定義する Bourne シェルスクリプトで、rules ファイル内で指定し、Solaris ソフトウェアがシステムにインストールされた後、システムがリブートする前に作業を実行します。finish スクリプトは、カスタム JumpStart インストールで Solaris をインストールするときのみ使用できます。

finish スクリプトは、次のような作業に利用できます。

- ファイルの追加
- パッケージまたはパッチの追加
- ルート環境のカスタマイズ
- システムのルートパスワードの設定
- 追加ソフトウェアのインストール

finish スクリプトに関する注意事項

- Solaris suninstall プログラムは、システムのファイルシステムを /a にマウントします。ファイルシステムは、システムがリブートするまで /a にマウントされています。したがって、/a にマウントされている新しくインストールされたファイルシステムの階層にファイルを追加、変更、または削除するように finish スクリプトを作成できます。
- finish スクリプトの出力は、/var/sadm/finish.log に保存されます。
- finish スクリプトの所有者は root、アクセス権は 644 にしてください。
- finish スクリプト内では、カスタム JumpStart 環境変数を使用できます。環境変数の一覧は、226 ページの「カスタム JumpStart の環境変数」を参照してください。
- finish スクリプトは、JumpStart ディレクトリに保存してください。

▼ finish スクリプトでファイルを追加する方法

finish スクリプトにより、JumpStart ディレクトリにあるファイルをインストールされたシステムへ追加できます。これは、SI_CONFIG_DIR 変数で指定されるディレクトリ (デフォルトは /tmp/install_config) に JumpStart ディレクトリがマウントされるためです。

注 – インストールされたシステムにすでに存在するファイルに、JumpStart ディレクトリからファイルをコピーして、ファイルを置換することもできます。

1. インストールされたシステムに追加するすべてのファイルを **JumpStart** ディレクトリにコピーします。
2. 新しくインストールされたファイルシステムの階層にコピーしたいファイルごとに、次の行を **finish** スクリプトに挿入します。

```
cp ${SI_CONFIG_DIR}/file_name /a/path_name
```

たとえば、自分のサイトのすべてのユーザーを対象に開発された特別なアプリケーション `site_prog` があると仮定します。 `site_prog` のコピーを JumpStart ディレクトリに置いた場合、 `finish` スクリプト内の次の行は JumpStart ディレクトリからシステムの `/usr/bin` ディレクトリに `site_prog` をコピーします。

```
cp ${SI_CONFIG_DIR}/site_prog /a/usr/bin
```

finish スクリプトによるパッケージとパッチの追加

`finish` スクリプトを作成すると、Solaris ソフトウェアをシステムにインストールした後に、パッケージやパッチを自動的に追加できます。 `finish` スクリプトを使用してパッケージを追加すると時間を節約できるだけでなく、サイトのさまざまなシステムにパッケージおよびパッチをインストールする上での整合性を維持できます。

`pkgadd(1M)` コマンドや `patchadd(1M)` コマンドを `finish` スクリプトで使用するときは、 `-R` オプションを使用して、 `/a` をルートパスとして指定する必要があります。

例 24-2 はパッケージを追加する `finish` スクリプトの例を示します。

例 24-2 `finish` スクリプトでパッケージを追加する

```
#!/bin/sh

BASE=/a
MNT=/a/mnt
ADMIN_FILE=/a/tmp/admin

mkdir ${MNT}
mount -f nfs sherlock:/export/package ${MNT}1
cat>${ADMIN_FILE} <<DONT_ASK2
mail=root
instance=overwrite
partial=nocheck
runlevel=nocheck
idepend=nocheck
rdepend=nocheck
space=ask
setuid=nocheck
```


例 24-2 finish スクリプトでパッケージを追加する (続き)

```
conflict=nocheck
action=nocheck
basedir=default
DONT_ASK

/usr/sbin/pkgadd -a ${ADMIN_FILE} -d ${MNT} -R ${BASE} SUNWxyz3
umount ${MNT}
rmdir ${MNT}
```

1. インストールするパッケージを含むサーバー上にディレクトリをマウントします。
2. 一時的なパッケージ管理ファイル `admin` を作成し、パッケージのインストール時に `pkgadd (1M)` コマンドがチェックも質問の表示も行わないようにします。この一時的なパッケージ管理ファイルは、パッケージの追加時に自動インストールを実施するために使用してください。
3. `-a` オプション (パッケージ管理ファイルを指定) と `-R` オプション (ルートパスを指定) を使用してパッケージを追加します。

注 – 以前は `finish` スクリプト環境では、`pkgadd` コマンドや `patchadd` コマンドとともに `chroot (1M)` コマンドが使用されていました。しかし、一部のパッケージやパッチには `-R` オプションが正しく動作しないものがあります。このような状況では、`chroot` コマンドを実行する前に `/a` ルートパスにダミーの `/etc/mnttab` ファイルを作成する必要があります。

`/etc/mnttab` ダミーファイルを作成するには、次の行を `finish` スクリプトに追加します。

```
cp /etc/mnttab /a/etc/mnttab
```

finish スクリプトによるルート環境のカスタマイズ

`finish` スクリプトを使って、システムにインストールされたファイルをカスタマイズできます。たとえば、例 24-3 の `finish` スクリプトは、ルートディレクトリ内の `.cshrc` ファイルに情報を追加することによって、ルート環境をカスタマイズします。

例 24-3 finish スクリプトによるルート環境のカスタマイズ

```
#!/bin/sh
#
# Customize root's environment
#
echo "***adding customizations in /.cshrc"
test -f a/.cshrc || {
cat>> a/.cshrc <<EOF
```

例 24-3 finish スクリプトによるルート環境のカスタマイズ (続き)

```
set history=100 savehist=200 filec ignoreeof prompt="\$user@`uname -n`> "  
alias cp cp -i  
alias mv mv -i  
alias rm rm -i  
alias ls ls -FC  
alias h history  
alias c clear  
unset autologout  
EOF  
}
```

finish スクリプトによるシステムのルートパスワードの設定

Solaris ソフトウェアがシステムにインストールされると、そのシステムはリブートします。ブートプロセス終了前に、システムはルートパスワードを入力するように求めてきます。パスワードを入力するまで、システムはブート処理を終了できません。

set_root_pw という finish スクリプトは、auto_install_sample ディレクトリに保存されます。この finish スクリプトは、プロンプトを表示することなく root パスワードを自動的に設定する方法を示します。例 24-4 に set_root_pw を示します。

例 24-4 finish スクリプトによるシステムのルートパスワードの設定

```
#!/bin/sh  
#  
#      @(#)set_root_pw 1.4 93/12/23 SMI  
#  
# This is an example Bourne shell script to be run after installation.  
# It sets the system's root password to the entry defined in PASSWD.  
# The encrypted password is obtained from an existing root password entry  
# in /etc/shadow from an installed machine.  
  
echo "setting password for root"  
  
# set the root password  
PASSWD=dK05IBkSF42lw  
#create a temporary input file1  
cp /a/etc/shadow /a/etc/shadow.orig2  
  
mv /a/etc/shadow /a/etc/shadow.orig  
nawk -F: '{  
    if ( $1 == "root" )3  
        printf"%s:%s:%s:%s:%s:%s:%s:%s:%s\n", $1,passwd,$3,$4,$5,$6,$7,$8,$9  
    else  
        printf"%s:%s:%s:%s:%s:%s:%s:%s:%s\n", $1,$2,$3,$4,$5,$6,$7,$8,$9  
}' passwd="$PASSWD" /a/etc/shadow.orig> /a/etc/shadow  
#remove the temporary file  
rm -f /a/etc/shadow.orig4
```

例 24-4 finish スクリプトによるシステムのルートパスワードの設定 (続き)

```
# set the flag so sysidroot won't prompt for the root password
sed -e 's/0 # root/1 # root/' ${SI_SYS_STATE}> /tmp/state.$$5
mv /tmp/state.$$ ${SI_SYS_STATE}
```

1. PASSWD 変数に、システムの /etc/shadow ファイルの既存のエントリから取得した暗号化されたルートパスワードを設定します。
2. /a/etc/shadow の一時入力ファイルを作成します。
3. \$PASSWD をパスワードフィールドとして使用して、新しくインストールしたシステム用の /etc/shadow ファイルにあるルートエントリを変更します。
4. 一時的な /a/etc/shadow ファイルを削除します。
5. 状態ファイルのエントリを 0 から 1 へ変更します。これによりユーザーはルートパスワードの入力を求められません。この状態ファイルには、SI_SYS_STATE 変数 (現在の値は /a/etc/.sysIDtool.state) を使用してアクセスします。この値の変更によってスクリプトで問題が発生することを防ぐには、必ず \$SI_SYS_STATE を使用してこのファイルを参照してください。ここに示されている sed コマンドでは、0 と 1 の後にタブ文字が入っています。

注 – finish スクリプトを使用してシステムのルートパスワードを設定した場合は、finish スクリプト内の暗号化されたパスワードからルートパスワードを解読されないように対策を講じてください。

finish スクリプトを使用して Web Start インストールプログラムでソフトウェアをインストールする

finish スクリプトを使用すると、Solaris オペレーティング環境のインストールに続いてさらにソフトウェアをインストールできます。一部のソフトウェアプログラムは、インストール時に情報入力を求めるプロンプトを表示する Solaris Web Start プログラムでインストールできます。自動インストールを行うには、`-nodisplay` または `-noconsole` オプションを指定して Solaris Web Start プログラムを実行します。

表 24-1 Solaris Web Start オプション

オプション	説明
<code>-nodisplay</code>	GUI を使用せずにインストーラを実行します。 <code>-locales</code> オプションによってインストールが変更された場合を除き、デフォルトの製品インストールを使用してください。

表 24-1 Solaris Web Start オプション (続き)

オプション	説明
-noconsole	対話形式のテキストコンソールデバイスを介さずにインストールを実行します。UNIX スクリプトで <code>-nodisplay</code> と共に使用すると便利です。

詳細は、`installer(1M)` のマニュアルページを参照してください。

圧縮された構成ファイルの作成

`add_install_client` コマンドを使用してカスタム JumpStart 構成ファイルの場所を指定する代わりに、システムのブート時に構成ファイルの場所を指定することができます。ただし、`boot` コマンドに指定できるファイル名は1つだけです。したがって、すべてのカスタム JumpStart 構成ファイルを1つのファイルに圧縮する必要があります。圧縮された構成ファイルの形式は、次のいずれかにしてください。

- tar
- compress で圧縮された tar
- zip
- bzip で圧縮された tar

▼ 圧縮された構成ファイルを作成する方法

1. プロファイルサーバー上の **JumpStart** ディレクトリに移動します。

```
# cd jumpstart_dir_path
```

2. 圧縮ツールを使って、すべてのカスタム **JumpStart** 構成ファイルを1つのファイルに圧縮します。

注 - 圧縮構成ファイルに相対パスを含めないでください。カスタム JumpStart 構成ファイルは、圧縮されたファイルと同じディレクトリに置かれている必要があります。

圧縮された構成ファイルには、次のファイルが含まれている必要があります。

- プロファイル
- rules
- rules.ok

圧縮された構成ファイルに、さらに `sysidcfg` ファイルを入れることもできます。

3. 圧縮された構成ファイルを **NFS** サーバー、**HTTP** サーバー、またはローカルハードディスク上に保存します。

圧縮された構成ファイルの例

次の例は、`tar` コマンドを使って、圧縮された構成ファイルを `config.tar` という名前で作成します。カスタム JumpStart 構成ファイルはすべて `/jumpstart` ディレクトリにあります。

例 24-5 圧縮された構成ファイルの作成

```
# cd /jumpstart
# tar -cvf config.tar *
a profile 1K
a rules 1K
a rules.ok 1K
a sysidcfg 1K
```

ディスク構成ファイルの作成

この節では、単一ディスク構成ファイルまたは複数ディスク構成ファイルを作成する方法について説明します。ディスク構成ファイルを作成すると、単一のシステムから `pfinstall(1M)` を使用して、異なるディスク構成に対してプロファイルをテストできます。

▼ ディスク構成ファイルを作成する方法

1. テストしたいディスクを持つシステムを決定します。
2. スーパーユーザーになります。
3. 単一ディスク構成ファイルを作成するには、`prtvtoc(1M)` コマンドの出力をファイルに保存します。

```
# prtvtoc /dev/rdisk/device_name>disk_config_file
```

<code>/dev/rdisk/device_name</code>	システムのディスクのデバイス名。 <code>device_name</code> は、 <code>cwtxdys2</code> または <code>cxdys2</code> の形式にしてください。
<code>disk_config_file</code>	ディスク構成ファイルの名前

4. 複数のディスクに Solaris ソフトウェアをインストールするテストを

- 実行しない場合は、ここで作業は終了です。
- 実行する場合は、各ディスク構成ファイルを結合し、その出力を新しいファイルに保存します。

```
# cat disk_file1 disk_file2>multi_disk_config
```

新しいファイルは、次に示すように複数ディスク構成ファイルになります。

```
# cat 104_disk2 104_disk3 104_disk5>multi_disk_test
```

5. 前の手順で作成した複数のディスク構成ファイルにおいて、ディスクデバイス名のターゲット番号がそれぞれ

- 固有である場合は、ここで作業は終了です。
- 固有でない場合は、テキストエディタでファイルを開き、ターゲット番号を固有のものにします。

たとえば、次に示すように、異なるディスクデバイス名について同じターゲット番号 (t0) がファイルに含まれているとします。

```
* /dev/rdsk/c0t0d0s2 partition map
```

```
...
```

```
* /dev/rdsk/c0t0d0s2 partition map
```

この場合は次のように、2番目のターゲット番号を t2 に設定します。

```
* /dev/rdsk/c0t0d0s2 partition map
```

```
...
```

```
* /dev/rdsk/c0t2d0s2 partition map
```

ディスク構成ファイルの例

次の例では、104M バイトのディスクを持つシステムで、単一ディスク構成ファイル `104_test` を作成します。

例 24-6 ディスク構成ファイルの作成

`prtvtoc` コマンドの出力を `104_test` という単一ディスク構成ファイルに保存します。

```
# prtvtoc /dev/rdsk/c0t3d0s2>104_test
```

`104_test` ファイルは、次のようになります。

```
* /dev/rdsk/c0t3d0s2 partition map
```

```
*
```

例 24-6 ディスク構成ファイルの作成 (続き)

```
* Dimensions:
*   512 bytes/sector
*   72 sectors/track
*   14 tracks/cylinder
*   1008 sectors/cylinder
*   2038 cylinders*   2036 accessible cylinders
* Flags:
*   1: unmountable
*   10: read-only
*
*
*   First      Sector      Last
* Partition  Tag  Flags      Sector      Count      Sector  Mount Directory
*   1         2    00         0          164304     164303  /
*   2         5    00         0          2052288    2052287
*   3         0    00        164304     823536     987839  /disk2/b298
*   5         0    00        987840     614880     1602719 /install/298/sparc/work
*   7         0    00       1602720    449568     2052287 /space
```

これで、ディスク構成ファイルは作成されました。ディスク構成ファイルを使用したプロファイル进行测试する方法の詳細は、159 ページの「プロファイルのテスト」を参照してください。

サイト固有のインストールプログラムの使用

`begin` スクリプトと `finish` スクリプトを使用すると、独自のインストールプログラムにより Solaris ソフトウェアをインストールできます。

プロファイルフィールドにマイナス記号 (-) が指定されているときには、`begin` スクリプトと `finish` スクリプトが、プロファイルや Solaris `suninstall` プログラムの代わりに、システムのインストール方法を制御します。

たとえば次のルールに一致した場合は、`begin` スクリプト `x_install.beg` と `finish` スクリプト `x_install.fin` が、`clover` という名前のシステムに Solaris ソフトウェアをインストールします。

```
hostname clover x_install.beg - x_install.fin
```


第 25 章

カスタムルールおよびプローブキーワードの作成

この章では、独自のカスタムルールとプローブキーワードを作成するための情報と手順を示します。

- 177 ページの「プローブキーワード」
- 178 ページの「custom_probes ファイルの作成」
- 181 ページの「custom_probes ファイルの妥当性検査」

プローブキーワード

プローブキーワードを理解するには、まずルールキーワードが何であるかを思い出してください。ルールキーワードは、ホスト名 (hostname) やメモリーサイズ (memsize) などの一般的なシステム属性を記述する事前定義された字句単位または語です。ルールキーワードとそれに関連する値を使用すると、同じ属性を持つシステムをプロファイルに対応付けることができます。システム属性を対応させるこの方法では、グループ内の各システムに Solaris ソフトウェアをどのようにインストールするかが定義されます。

カスタム JumpStart の環境変数は、begin スクリプトおよび finish スクリプトで使用しますが、必要に応じて設定します。たとえば、すでに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に関する情報を入手できるのは、installed ルールキーワードに続いて SI_INSTALLED を使用する場合だけです。

場合によっては、システムの照合とプロファイルの実行以外の目的で、この同じ情報を begin および finish スクリプトで抽出する必要があります。プローブキーワードは、この解決方法となります。プローブキーワードは属性情報を抽出するので、照合条件を設定してプロファイルを実行する必要がなくなります。

プローブキーワードと値の一覧は、228 ページの「プローブキーワードと値」を参照してください。

custom_probes ファイルの作成

197 ページの「ルールキーワードと値の説明」と 228 ページの「プローブキーワードと値」で説明したルールおよびプローブキーワードでは不十分な場合は、`custom_probes` ファイルを作成して、独自のカスタムルールまたはプローブキーワードを定義できます。

`custom_probes` ファイルは、2 種類のタイプの関数を含む Bourne シェルスクリプトです。`custom_probes` ファイルは、`rules` ファイルを保存したのと同じ `JumpStart` ディレクトリに保存する必要があります。次に、`custom_probes` ファイルに定義できる 2 種類の関数を示します。

- **プローブ** – 必要な情報を収集するかあるいは実際の作業を実行し、定義された対応する `SI_` 環境変数を設定します。プローブ関数はプローブキーワードになります。
- **比較** – 対応するプローブ関数を呼び出してプローブ関数の出力を比較し、キーワードが一致する場合は 0、キーワードが一致しない場合は 1 を返します。比較関数はルールキーワードになります。

custom_probes ファイルの構文

`custom_probes` ファイルには、任意の有効な Bourne シェルコマンド、変数、またはアルゴリズムを含めることができます。

注 – `custom_probes` ファイルで単一の引数を必要とするプローブおよび比較関数を定義できます。`rules` ファイル内で対応するカスタムプローブキーワードを使用する場合、このキーワードの後に続く引数が (`$1` として) 解釈されます。

`rules` ファイル内で対応するカスタムルールキーワードを使用する場合、引数はこのキーワードの後から解釈され、次の `&&` または `begin` スクリプトのどちらか最初にきた方の前で解釈が終了します。

`custom_probes` ファイルは、次の要件を満たす必要があります。

- `custom_probes` という名前を使用する
- 所有者が `root` である
- 実行可能であり、アクセス権が 755 に設定されている
- 少なくとも 1 つのプローブ関数と対応する比較関数を含む

より明確で効果的な構成にするために、最初にすべてのプローブ関数をファイルの一番上に定義してから、比較関数を続けて定義してください。

custom_probes の関数名の構文

プローブ関数名は、probe_ で始める必要があります。比較関数名は、cmp_ で始める必要があります。

probe_ で始まる関数は、新しいプローブキーワードを定義します。たとえば、関数 probe_tcx は、新しいプローブキーワード tcx を定義します。cmp_ で始まる関数は、新しいルールキーワードを定義します。たとえば、cmp_tcx は、新しいルールキーワード tcx を定義します。

▼ custom_probes ファイルを作成する方法

1. テキストエディタを使用して **Bourne** シェルスクリプトのテキストファイルを作成します。このファイルに custom_probes という名前を付けます。
2. custom_probes テキストファイルで、プローブ関数と比較関数を定義します。

注 - custom_probes ファイルには、引数を必要とするプローブおよび比較関数を定義できます。rules ファイル内で対応するカスタムプローブキーワードを使用する場合、このキーワードの後に続く引数は順番に (\$1, \$2 などとして) 解釈されます。

rules ファイル内で対応するカスタムルールキーワードを使用する場合、引数はこのキーワードの後から解釈され、次の && または begin スクリプトのどちらか最初に来た方の前で解釈が終了します。

3. custom_probes ファイルを **JumpStart** ディレクトリに保存します (rules ファイルと同じ)。
4. root が rules ファイルを所有していて、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。

custom_probes ファイルとキーワードの例

プローブ関数と比較関数の例は次のディレクトリにも入っています。

- Solaris ソフトウェアがインストールされたシステムの /usr/sbin/install.d/chkprobe
- Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD 上の /Solaris_9/Tools/Boot/usr/sbin/install.d/chkprobe

次の custom_probes ファイルには、TCX グラフィックスカードがあるかどうかをテストするプローブおよび比較関数が含まれます。

例 25-1 custom_probes ファイル

```
#!/bin/sh
#
# custom_probe script to test for the presence of a TCX graphics card.
#
#
# PROBE FUNCTIONS
#
probe_tcx() {
    SI_TCX=`modinfo | grep tcx | nawk '{print $6}'`
    export SI_TCX
}

#
# COMPARISON FUNCTIONS
#
cmp_tcx() {
    probe_tcx

    if [ "X${SI_TCX}" = "X${1}" ]; then
        return 0
    else
        return 1
    fi
}
}
```

次のサンプル rules ファイルは、前の例で定義されているプローブキーワード (tcx) の使用方法を示しています。TCX グラフィックスカードがシステムにインストールされていて検出されると、profile_tcx が実行されます。そうでない場合は、profile が実行されます。

注 - プローブキーワードは、rules ファイルの先頭または先頭近くに指定してください。これは、そのプローブキーワードに依存する可能性のあるほかのルールキーワードよりも前に読み取られて実行されるようにするためです。

例 25-2 rules ファイルで使用されるカスタムプローブキーワード

```
probe tcx
tcx    tcx    -    profile_tcx    -
any    any    -    profile        -
```

custom_probes ファイルの妥当性検査

プロファイル、rules、および custom_probes ファイルを使用するには、これらのファイルの設定が正しいかどうかを check スクリプトを実行して検査する必要があります。すべてのプロファイル、ルール、およびプローブ関数と比較関数が正しく設定されている場合は、rules.ok ファイルと custom_probes.ok ファイルが作成されます。表 25-1 は、check スクリプトの実行内容を説明しています。

表 25-1 check スクリプトを使用した場合の動作

手順	説明
1	check は、custom_probes ファイルを検索します。
2	このファイルが存在する場合、check は custom_probes ファイルから custom_probes.ok ファイルを作成し、すべてのコメントと空白行を削除し、すべての Bourne シェルコマンド、変数、およびアルゴリズムを保存して、次に check は次のコメント行をファイルの最後に追加します。 # version=2 checksum=num

▼ custom_probes ファイルを妥当性検査する方法

1. check スクリプトが **JumpStart** ディレクトリにあることを確認します。

注 – check スクリプトは、Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD の Solaris_9/Misc/jumpstart_sample ディレクトリにあります。

2. **JumpStart** ディレクトリに移動します。
3. check スクリプトを実行して、rules ファイルおよび custom_probes ファイルの妥当性を検査します。

```
$ ./check [-p path -r file_name]
```

- p *path*** 使用しているシステムの check スクリプトではなく、プラットフォームの Solaris 9 ソフトウェアイメージの check スクリプトを使用して、`custom_probes` ファイルを妥当性検査します。*path* は、ローカルディスクか、あるいはマウントされた Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD 上のイメージです。
- システムが以前のバージョンの Solaris を実行している場合、このオプションを使用して、一番新しい check バージョンを実行します。
- r *file_name*** `custom_probes` 以外のファイル名を指定します。-r オプションを使用することで、一連の関数を `custom_probes` ファイル内にまとめる前にそれらの妥当性検査を実施できます。

check スクリプトを実行すると、`rules` ファイルと `custom_probes` ファイル、および各プロファイルの妥当性が表示されます。エラーが検出されないと、“The custom JumpStart configuration is ok” と表示され、JumpStart ディレクトリ内に `rules.ok` ファイルと `custom_probes.ok` ファイルが作成されます。

4. `custom_probes.ok` ファイルが
 - 実行可能な場合は、手順 5 へ進みます。
 - 実行可能でない場合は、次のコマンドを入力します。
chmod +x custom_probes
5. `root` が `custom_probes.ok` ファイルを所有し、そのアクセス権が **755** に設定されていることを確認してください。

第 26 章

カスタム JumpStart インストールの実行

この章では、カスタム JumpStart インストールの実行方法について説明します。これらの手順は、Solaris 9 ソフトウェアをインストールしようとするシステム上で実行してください。

カスタム JumpStart インストールの実行

作業マップ: カスタム JumpStart インストールのためのシステムの設定

カスタム JumpStart インストールの際に、JumpStart プログラムはインストール対象のシステムを `rules.ok` ファイル内のルールに対応付けます。この際、最初から順にルールが読み取られ、インストール中のシステムがルールに定義されたすべてのシステム属性に一致するかどうか検査されます。システムがルールと一致すると同時に、JumpStart プログラムによる `rules.ok` ファイルの読み取りが停止して、一致したルールのプロファイルに基づいてシステムのインストールが開始されます。

表 26-1 作業マップ: カスタム JumpStart インストールのためのシステムの設定

作業	説明	参照先
システムがサポートされているかを確認する	ハードウェアのマニュアルで、Solaris 9 環境におけるシステムサポートを確認します。	『Solaris 9 Sun ハードウェアマニュアル』

表 26-1 作業マップ: カスタム JumpStart インストールのためのシステムの設定 (続き)

作業	説明	参照先
Solaris ソフトウェアをインストールするための十分なディスク容量がシステムにあるかを確認する	Solaris ソフトウェアのインストールに求められる十分な容量があるかを確認します。	第 5 章
(省略可能) システム構成情報を事前設定する	sysidcfg ファイルまたはネームサービスを使用してシステムのインストール情報をあらかじめ設定できます。システム情報をあらかじめ設定すると、インストール時に情報入力を求めるプロンプトは表示されません。	第 7 章
カスタム JumpStart インストールするようにシステムを準備する	rules ファイルとプロファイルファイルを作成して検証します。	第 23 章
(省略可能) カスタム JumpStart のオプション機能を準備する	begin スクリプト、finish スクリプトなどのオプション機能を使用する場合は、それらのスクリプトまたはファイルを用意します。	第 24 章 と 第 25 章
(省略可能) ネットワーク上でインストールするためのシステムを設定する	リモートの Solaris 9 DVD または Solaris 9 SOFTWARE CD イメージをインストールする場合は、インストールサーバーまたはブートサーバーからシステムをブートしてインストールするように、システムを設定する必要があります。	第 12 章
インストールまたはアップグレードを行う	システムをブートしてインストールまたはアップグレードを開始します。	184 ページの「カスタム JumpStart プログラムを使用してインストールまたはアップグレードする方法」

▼ カスタム JumpStart プログラムを使用してインストールまたはアップグレードする方法

1. システムがネットワークに接続されている場合は、**Ethernet** コネクタまたはそれに類似したネットワークアダプタがシステムに装着されていることを確認します。
2. tip(1) ラインで接続されているシステムをインストールする場合は、ウィンドウ表示が横 **80** 桁、縦 **24** 行以上あることを確認します。

stty(1) コマンドを使うと、tip ウィンドウの現在のサイズを調べることができます。

3. システムの **DVD-ROM** または **CD-ROM** ドライブを使用して **Solaris 9** ソフトウェアをインストールする場合は、ドライブに **Solaris 9 DVD** または **Solaris 9 SOFTWARE 1 of 2 CD** を挿入します。
4. プロファイルフロッピーディスクを使用する場合は、システムのプロロッピーディスクドライブにそのフロッピーディスクを挿入します。
5. システムをブートします。
 - 新しく購入したばかり (未使用) のシステムの場合は、システムの電源を入れます。
 - 既存システムのインストールまたはアップグレードを行う場合は、システムをシャットダウンします。ok プロンプトで次のコマンドを入力します。

```
ok boot cdrom:net - install [url:ask] [dhcp] [nowin]
```

cdrom	CD または DVD からブートすることを指定します。 システムの EEPROM が旧式のものである場合は、cdrom を sd(0,6,2) に置き換えてシステムの CD-ROM または DVD-ROM ドライブからブートしてください。
net	ネットワーク上のインストールサーバーからブートすることを意味します。

<i>url</i>	<p>カスタム JumpStart ファイルの場所を指定します。以下にあるファイルを URL で指定できます。</p> <ul style="list-style-type: none"> ■ ローカルハードディスク <ul style="list-style-type: none"> <code>file://jumpstart_dir_path/compressed_config_file</code> ■ ネットワークファイルシステム (NFS) サーバー <ul style="list-style-type: none"> <code>nfs://server_name:IP_address/jumpstart_dir_path/compressed_config_file</code> ■ HTTP サーバー <ul style="list-style-type: none"> <code>http://server_name:IP_address/jumpstart_dir_path/compressed_config_file&proxy_info</code> <p>sysidcfg ファイルを圧縮構成ファイルに含めた場合は、次の例に示すようにそのファイルを含むサーバーの IP アドレスを指定する必要があります。</p> <p><code>http://131.141.2.32/jumpstart/config.tar</code></p> <p>圧縮構成ファイルをファイアウォールで防御された HTTP サーバー上の圧縮構成ファイルに保存した場合は、ブート時に proxy 指示子を使用する必要があります (ファイルが配置されたサーバーの IP アドレスを指定する必要はない)。プロキシサーバーの IP アドレスは次のように指定します。</p> <p><code>http://www.shadow.com/jumpstart/config.tar&proxy=131.141.6.151</code></p>
<i>ask</i>	<p>システムが起動し、ネットワークへの接続が完了すると、インストールプログラムから圧縮構成ファイルの場所を入力するプロンプトが表示されます。</p> <p>Return キーを押してこのプロンプトへの入力を行わないと、インストールプログラムはネットワークパラメータを対話的に構成し、圧縮構成ファイルの場所を尋ねるプロンプトを表示します。Return キーを押してこのプロンプトへの入力を行わないと、Solaris suninstall プログラムが開始されます。</p>
<i>dhcp</i>	<p>DHCP サーバーからシステムのブートに必要なネットワークインストール情報を取得します。</p> <p>dhcp を指定しないと、<code>/etc/bootparams</code> ファイル、またはネームサービスの <code>bootparams</code> データベースが使用されます。</p>
<i>nowin</i>	<p>X Window を起動しません。カスタム JumpStart インストールの実行には、X Window は必要ありません。このため、このオプションを指定すると、インストール時間を短縮できます。</p>

注 - ハードウェアおよびシステムコンポーネントがシステムによって検査されて、システムがブートします。ブートには数分間かかります。

6. `sysidcfg` でシステム情報の事前構成を行わなかった場合は、プロンプトが表示された時点でシステム構成についての質問に答えます。
7. 画面の指示に従って、ソフトウェアをインストールします。
JumpStart プログラムが Solaris ソフトウェアのインストールを終了すると、システムが自動的にリブートします。
インストールが終了すると、インストールログがファイルに保存されます。インストールログは、次のディレクトリに作成されます。
 - `/var/sadm/system/logs`
 - `/var/sadm/install/logs`

第 27 章

カスタム JumpStart によるインストール例

この章では、カスタム JumpStart インストールによる Solaris ソフトウェア の設定およびインストール例を紹介します。

- 189 ページの「サイトの設定例」
- 191 ページの「インストールサーバーの作成」
- 192 ページの「マーケティングシステム用のブートサーバーの作成」
- 192 ページの「JumpStart ディレクトリの作成」
- 193 ページの「JumpStart ディレクトリの共有」
- 193 ページの「エンジニアリンググループのプロファイルの作成」
- 194 ページの「マーケティンググループのプロファイルの作成」
- 194 ページの「rules ファイルの更新」
- 195 ページの「rules ファイルの妥当性を検査する」
- 195 ページの「ネットワーク上でインストールするためのエンジニアリングシステムの設定」
- 196 ページの「ネットワーク上でインストールするためのマーケティングシステムの設定」
- 196 ページの「システムのブートと Solaris 9 ソフトウェアのインストール」

サイトの設定例

この例で使用するサイトの設定を図 27-1 に示します。

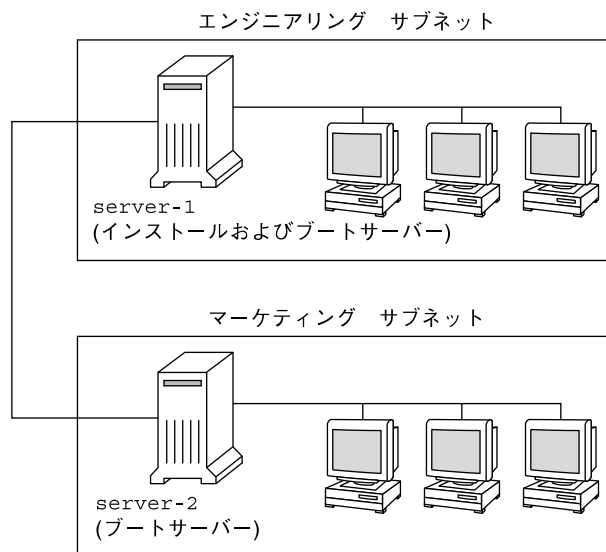


図 27-1 サイトの設定例

この例におけるサイトの状況は次のとおりです。

- エンジニアリンググループは、グループ固有のサブネット上にあります。このグループでは、ソフトウェア開発を行うためにシステムを使用しています。
- マーケティンググループも、グループ固有のサブネット上にあります。このグループでは、文書処理、表計算、およびその他のオフィスツールを実行するためにシステムを使用しています。
- このサイトでは NIS を使用しています。システムの Ethernet アドレス、IP アドレス、およびホスト名は NIS マップに事前設定されています。サイトのサブネットマスク、日付と時刻、および地域の情報も NIS マップに事前設定されています。

注 - マーケティングシステムの周辺装置は、`sysidcfg` ファイルに事前設定されています。

- エンジニアリングシステムとマーケティングシステムの両方とも、Solaris 9 ソフトウェアをネットワーク上でインストールします。

インストールサーバーの作成

これらのグループは Solaris ソフトウェアをネットワーク上でインストールする必要があるため、server-1 を両方のグループのインストールサーバーにします。server-1 ローカルディスク (/export/install ディレクトリ内) にイメージをコピーするには、setup_install_server(1M) コマンドを使用します。これらのイメージは、Solaris 9 SOFTWARE CD と Solaris 9 LANGUAGES CD、または Solaris 9 DVD からコピーしてください。

イメージは必ず空のディレクトリにコピーしてください。以下の例では、sparc_9 ディレクトリにコピーします。

例 27-1 Solaris 9 CD のコピー

server-1 の CD-ROM ドライブに Solaris 9 SOFTWARE 1 of 2 CD を挿入し、次のコマンドを入力します。

```
server-1# mkdir -p /export/install/sparc_9
server-1# cd /CD_mount_point/Solaris_9/Tools
server-1# ./setup_install_server /export/install/sparc_9
```

server-1 の CD-ROM ドライブに Solaris 9 SOFTWARE 2 of 2 CD を挿入し、次のコマンドを入力します。

```
server-1# cd /CD_mount_point/Solaris_9/Tools
server-1# ./add_to_install_server /export/install/sparc_9
```

server-1 の CD-ROM ドライブに Solaris 9 LANGUAGES CD を挿入し、次のコマンドを入力します。

```
server-1# cd /CD_mount_point/Solaris_9/Tools
server-1# ./add_to_install_server /export/install/sparc_9
```

例 27-2 Solaris 9 DVD のコピー

server-1 の DVD-ROM ドライブに Solaris 9 DVD を挿入し、次のコマンドを入力します。

```
server-1# mkdir -p /export/install/sparc_9
server-1# cd /DVD_mount_point/Solaris_9/Tools
server-1# ./setup_install_server /export/install/sparc_9
```

マーケティングシステム用のブートサーバーの作成

システムは、異なるサブネット上のインストールサーバーからはブートできません。したがって、server-2 をマーケティンググループのサブネット上のブートサーバーにします。setup_install_server(1M) コマンドを使用して Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD から server-2 ローカルディスクの /export/boot ディレクトリへブートソフトウェアをコピーします。

server-2 の CD-ROM ドライブに Solaris 9 SOFTWARE 1 of 2 CD を挿入する場合は、次のコマンドを入力してください。

```
server-2# cd /CD_mount_point/Solaris_9/Tools
server-2# ./setup_install_server -b /export/boot
```

server-2 の DVD-ROM ドライブに Solaris 9 DVD を挿入する場合は、次のコマンドを入力してください。

```
server-2# cd /DVD_mount_point/Solaris_9/Tools
server-2# ./setup_install_server -b /export/boot
```

setup_install_server コマンドの -b は、setup_install_server が /export/boot というディレクトリにブート情報をコピーすることを指定します。

JumpStart ディレクトリの作成

インストールサーバーとブートサーバーの設定が終了したところで、次は server-1 に JumpStart ディレクトリを作成します (ネットワーク上の任意のシステムを利用できます)。このディレクトリには、Solaris ソフトウェアのカスタム JumpStart インストールに必要なファイルが入っています。このディレクトリを設定するには、/export/install にコピーされている Solaris 9 DVD イメージまたは Solaris 9 SOFTWARE 1 of 2 CD イメージからサンプルディレクトリをコピーします。

```
server-1# mkdir /jumpstart
server-1# cp -r /export/install/sparc_9/Solaris_9/Misc/jumpstart_sample /jumpstart
```

JumpStart ディレクトリの共有

ネットワーク上のシステムが rules ファイルやプロファイルにアクセスできるようにするために、/jumpstart ディレクトリを共有します。このためには、/etc/dfs/dfstab ファイルに次の行を追加します。

```
share -F nfs -o ro,anon=0 /jumpstart
```

次に、コマンド行で shareall コマンドを使用します。

```
server-1# shareall
```

エンジニアリンググループのプロファイルの作成

エンジニアリングシステムの場合、eng_prof ファイルを /jumpstart ディレクトリに作成します。eng_prof ファイルのエントリは次のようになっていて、エンジニアリンググループのシステムにインストールされる Solaris 9 ソフトウェアを定義しています。

```
install_type  initial_install1
system_type   standalone2
partitioning  default3
cluster       SUNWCprog4
fileysys      any 512 swap5
```

1. アップグレードではなく、初期インストールによりインストールするよう指定します。
2. エンジニアリングシステムが、スタンドアロンシステムであることを指定します。
3. JumpStart ソフトウェアが、デフォルトのディスクパーティションを使用して、エンジニアリングシステムに Solaris をインストールするよう指定します。
4. 開発者システムサポートソフトウェアグループをインストールするよう指定します。
5. エンジニアリンググループの各システムが、512M バイトのスワップ領域を持つよう指定します。

マーケティンググループのプロファイルの作成

マーケティングシステムの場合、`marketing_prof` ファイルを `/jumpstart` ディレクトリに作成します。`marketing_prof` ファイルのエントリは次のようになっています。マーケティンググループのシステムにインストールされる Solaris 9 ソフトウェアを定義しています。

```
install_type  initial_install1
system_type   standalone2
partitioning  default3
cluster       SUNWCuser4
package       SUNWaudio5
```

1. アップグレードではなく、初期インストールによりインストールするよう指定します。
2. マーケティングシステムが、スタンドアロンであることを指定します。
3. JumpStart ソフトウェアが、デフォルトのディスクパーティションを使用して、マーケティングシステムに Solaris をインストールするよう指定します。
4. 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をインストールするよう指定します。
5. オーディオツールのデモソフトウェアパッケージを各システムに追加するよう指定します。

rules ファイルの更新

次に、`rules` ファイルにルールを追加する必要があります。Solaris インストールプログラムは、カスタム JumpStart インストール中、これらのルールを使用して、各システムに正しいインストール (プロファイル) を選択します。

このサイトでは、各部署は独自のサブネットとネットワークアドレスを持っています。エンジニアリング部はサブネット 255.222.43.0 にあり、マーケティング部はサブネット 255.222.44.0 にあります。この情報を使用すれば、エンジニアリングシステムとマーケティングシステムのインストール方法を制御できます。`/jumpstart` ディレクトリ内の `rules` ファイルを編集して、サンプルのルールをすべて削除し、次の行をファイルに追加します。

```
network 255.222.43.0 - eng_prof -
network 255.222.44.0 - marketing_prof -
```

これらのルールは、基本的に、255.222.43.0 ネットワーク上のシステムが `eng_prof` プロファイルを使用してインストールされることと、255.222.44.0 ネットワーク上のシステムが `marketing_prof` プロファイルを使用してインストールされることを示しています。

注 - サンプルのルールでは、どのシステムが `eng_prof` または `marketing_prof` を使用してインストールされるかをネットワークアドレスを使用して識別できます。この他にも、ホスト名、メモリーサイズ、またはモデルタイプをルールのキーワードとして使用できます。`rules` ファイルで使用できるキーワードのリストについては、表 28-1 を参照してください。

rules ファイルの妥当性を検査する

`rules` ファイルとプロファイルを適切に設定した後は、`check` スクリプトを実行して、これらのファイルを確認します。

```
server-1# cd /jumpstart
server-1# ./check
```

`check` スクリプトによってエラーが検出されない場合は、`rules.ok` ファイルが作成されます。

ネットワーク上でインストールするためのエンジニアリングシステムの設定

`/jumpstart` ディレクトリおよび必要なファイルを設定した後は、インストールサーバー (`server-1`) で `add_install_client` コマンドを使用して、インストールサーバーから Solaris 9 ソフトウェアをインストールできるようにエンジニアリングシステムを設定します。`server-1` は、エンジニアリンググループのサブネットにとってはブートサーバーにも相当します。

```
server-1# cd /export/install/sparc_9/Solaris_9/Tools
server-1# ./add_install_client -c server-1:/jumpstart host-eng1 sun4u
server-1# ./add_install_client -c server-1:/jumpstart host-eng2 sun4u
```

`add_install_client` コマンドで使用されるオプションの意味は次のとおりです。

-c	サーバー (server-1) と JumpStart ディレクトリへのパス (/jumpstart) を指定する
host-eng1	エンジニアリンググループのシステム名
host-eng2	エンジニアリンググループの別のシステム名
sun4u	server-1 をインストールサーバーとして使用するシステムのアーキテクチャを指定する Ultra 5 システムのプラットフォームグループ

ネットワーク上でインストールするためのマーケティングシステムの設定

次に、ブートサーバー (server-2) で `add_install_client` コマンドを使用して、ブートサーバーからブートして、インストールサーバー (server-1) から Solaris 9 ソフトウェアをインストールするようにマーケティングシステムを設定します。

```
server-2# cd /marketing/boot-dir/Solaris_9/Tools
server-2# ./add_install_client -c server-1:/jumpstart host-mkt1 sun4u
server-2# ./add_install_client -c server-1:/jumpstart host-mkt2 sun4u
```

`add_install_client` コマンドで使用されるオプションの意味は次のとおりです。

-c	サーバー (server-1) と JumpStart ディレクトリへのパス (/jumpstart) を指定する
host-mkt1	マーケティンググループのシステム名
host-mkt2	マーケティンググループの別のシステム名
sun4u	server-1 をインストールサーバーとして使用するシステムのアーキテクチャを指定する Ultra 5 システムのプラットフォームグループ

システムのブートと Solaris 9 ソフトウェアのインストール

サーバーとファイルの設定が完了した後は、各システムの `ok (PROM)` プロンプトで次の `boot` コマンドを使用して、システムをブートできます。

```
ok boot net - install
```

システムに Solaris オペレーティング環境が自動的にインストールされます。

第 28 章

カスタム JumpStart に関するリファレンス情報

この章では、rules ファイル、プロファイル、および begin/finish スクリプトで使用できるキーワードと値のリストを示します。

- 197 ページの「ルールキーワードと値の説明」
- 202 ページの「プロファイルキーワードと値」
- 226 ページの「カスタム JumpStart の環境変数」
- 228 ページの「プローブキーワードと値」

ルールキーワードと値の説明

表 28-1 で、rules ファイルで使用できるルールキーワードとルール値について説明します。rules ファイルの作成方法についての詳細は、152 ページの「rules ファイルの作成」を参照してください。

表 28-1 ルールキーワードとルール値の説明

キーワード	値	説明
any	マイナス記号 (-)	あらゆるものに一致することを意味する。
arch	<i>processor_type</i> <i>processor_type</i> に有効な値は sparc	システムのプロセッサタイプを照合する。 システムのプロセッサタイプは、uname -p コマンドで調べることができる

表 28-1 ルールキーワードとルール値の説明 (続き)

キーワード	値	説明
disksize	<p><i>actual_disk_name size_range</i></p> <p><i>actual_disk_name</i> — cxydz (c0t3d0 など) という形式のディスク名、または特別なワード「rootdisk」。rootdisk を使用する場合、照合するディスクは次の順番で決定される。</p> <ul style="list-style-type: none"> ■ 事前にインストールされているブートイメージ (つまり、ファクトリ JumpStart でインストールされた新しいシステム) を持つディスク ■ c0t3d0s0 ディスク (存在する場合) ■ 最初に利用可能なディスク (カーネルのプロンプ順で検索される) <p><i>size_range</i> - ディスクのサイズ。M バイト単位の範囲 (x-x) で指定する必要がある</p>	<p>システムのディスクの名前とサイズを照合する (M バイト単位)。</p> <p>例:</p> <pre>disksize c0t3d0 250-300</pre> <p>この例では、JumpStart プログラムは c0t3d0 というシステムディスクの照合を試みる。このディスクは 250M から 300M バイトの情報を保持できる。</p> <p>例:</p> <pre>disksize rootdisk 750-1000</pre> <p>この例では、JumpStart プログラムはまず事前にインストールされたブートイメージが入ったシステムディスクを照合する。次に、c0t3d0s0 ディスク (存在する場合) を照合する。最後に、750M から 1G バイトの情報を保持できるディスクの中で利用可能な最初のディスクを照合する。</p> <p>注 - <i>size_range</i> を計算するときは、1M バイトが 1,048,576 バイトであることに注意してください。「535M バイト」ディスクと明記されているディスクでも、ディスク空間が 510M バイトしかない場合があります。535,000,000/1,048,576=510 により、JumpStart は「535M バイト」ディスクを実際には 510M バイトのディスクと見なします。したがって、この「535M バイト」ディスクは「530-550」の <i>size_range</i> には一致しません。</p>
domainname	<i>actual_domain_name</i>	<p>システムのドメイン名を照合する。ドメイン名でネームサービスが情報を判別する方法を制御する。</p> <p>システムがインストール済みの場合、domainname コマンドによりシステムのドメイン名を表示できる</p>
hostaddress	<i>actual_IP_address</i>	システムの IP アドレスを照合する

表 28-1 ルールキーワードとルール値の説明 (続き)

キーワード	値	説明
hostname	<i>actual_host_name</i>	システムのホスト名を照合する。 システムがインストール済みの場合、 <code>uname -n</code> コマンドによりシステムのホスト名を表示できる
installed	<i>slice version</i> <i>slice</i> - <i>cwtxdysz</i> 形式 (たとえば、 <i>c0t3d0s5</i>) のディスクスライス名、または <i>any</i> か <i>rootdisk</i> 。 <i>any</i> を使用すると、システムに接続されたどのディスクも照合する (カーネルのプロンプ順)。 <i>rootdisk</i> を使用する場合、照合するディスクは次の順番で決定される。 <ul style="list-style-type: none"> ■ 事前にインストールされているブートイメージ JumpStart でインストールされた新しいシステム) を持つディスク ■ <i>c0t3d0s0</i> ディスク (存在する場合) ■ 最初に利用可能なディスク (カーネルのプロンプ順で検索される) <i>version</i> - バージョン名、あるいは <i>any</i> または <i>upgrade</i> 。 <i>any</i> を使用すると、Solaris または SunOS リリースのどれとでも照合する。 <i>upgrade</i> を使用すると、アップグレード可能な Solaris 2.1 以降の互換リリースのどれとでも照合する。 Solaris リリースは検出されたが、バージョンを判断できないという場合は、返されるバージョンは <i>SystemV</i> になる。	Solaris ソフトウェアの特定バージョンに対応するルートファイルシステムが存在するディスクを照合する。 例: <pre>installed c0t3d0s1 Solaris_8</pre> この例では、 <i>c0t3d0s1</i> に Solaris 9 のルートファイルシステムを持つシステムを照合している
karch	<i>actual_platform_group</i> 有効な値は、 <i>sun4m</i> 、 <i>sun4u</i> 、 <i>prep</i> 。システムおよび対応するプラットフォームグループのリストは、『Solaris 9 Sun ハードウェアマニュアル』に記載されている。	システムのプラットフォームグループを照合する。 システムがインストール済みの場合は、 <code>arch -k</code> コマンドまたは <code>uname -m</code> コマンドにより、システムのプラットフォームグループを表示できる

表 28-1 ルールキーワードとルール値の説明 (続き)

キーワード	値	説明
memsize	<i>physical_mem</i> 値は M バイト単位の範囲 (x-x) または 1 つの M バイト値で指定する	システムの物理メモリーサイズを照合する (M バイト単位)。 例: memsize 64-128 この例では、64M から 128M バイトの物理メモリーサイズを持つシステムと照合している。 システムがインストール済みの場合は、prtconf コマンド (2 行目) によりシステムの物理メモリーサイズを表示できる
model	<i>actual_platform_name</i>	システムのプラットフォーム名を照合する。有効なプラットフォーム名については、『Solaris 9 Sun ハードウェアマニュアル』を参照してください。 インストール済みのシステムのプラットフォーム名を見つけるには、uname -i コマンドか prtconf コマンド (5 行目) の出力を使用する。 注 - <i>actual_platform_name</i> にスペースが含まれている場合は、スペースを下線 (_) で置き換える必要があります。 例: SUNW,Sun_4_50
network	<i>network_num</i>	システムのネットワーク番号を照合する。これは JumpStart プログラムが、システムの IP アドレスとサブネットマスクの論理積をとって判別する。 例: network 193.144.2.8 この例では、IP アドレスが 193.144.2.8 のシステムを照合する (サブネットマスクが 255.255.255.0 の場合)

表 28-1 ルールキーワードとルール値の説明 (続き)

キーワード	値	説明
osname	Solaris_x	<p>システムにすでにインストールされている Solaris のバージョンを照合する。</p> <p>例:</p> <pre>osname Solaris_9</pre> <p>この例では、JumpStart プログラムは Solaris 9 オペレーティング環境がすでにインストールされているシステムを照合する。</p>
probe	probe_keyword	<p>有効なプローブキーワードまたは有効なカスタムプローブキーワード</p> <p>例:</p> <pre>probe disks</pre> <p>この例は、システムのディスクのサイズ (M バイト) をカーネルプローブ順に返します (c0t3d0s1、c0t4d0s0 など)。JumpStart プログラムは環境変数 SI_DISKLIST、SI_DISKSIZE、SI_NUMDISKS、および SI_TOTALDISK を設定します。</p> <p>注 - probe キーワードには、属性を照合せず、プロファイルを実行しないという特徴があります。このキーワードは、値を返すだけです。したがって、probe ルールキーワードで、begin スクリプト、プロファイル、および finish スクリプトは指定できません。</p> <p>プローブキーワードについては、第 25 章を参照してください。</p>

表 28-1 ルールキーワードとルール値の説明 (続き)

キーワード	値	説明
totaldisk	<p><i>size_range</i></p> <p>値は M バイト単位の範囲 (<i>x-x</i>) で指定する必要があります</p>	<p>システムのディスク空間の全体量 (M バイト単位) を照合する。ディスク空間の全体量には、システムに接続されている使用可能なディスクがすべて含まれる。</p> <p>例:</p> <pre>totaldisk 300-500</pre> <p>この例では、全体として 300M から 500M バイトのディスク空間を持つシステムと照合している。</p> <p>注 - <i>size_range</i> を計算するとき、1M バイトが 1,048,576 バイトであることに注意してください。「535M バイト」ディスクと明記されているディスクでも、ディスク空間が 510M バイトしかない場合があります。535,000,000/1,048,576=510 により、JumpStart は「535M バイト」ディスクを実際には 510M バイトのディスクと見なします。したがって、この「535M バイト」ディスクは 530-550 の <i>size_range</i> には一致しません。</p>

プロファイルキーワードと値

次の節では、プロファイルで使用できるプロファイルキーワードとプロファイル値を説明します。プロファイル作成の詳細は、155 ページの「プロファイルの作成」を参照してください。

プロファイルキーワードのリスト

表 28-2 を使用すれば、どのキーワードがユーザーのインストールに適しているかを簡単に決定できます。プロファイルキーワードの説明で特に注記されていないかぎり、プロファイルキーワードは初期インストールオプションだけで使用できます。

表 28-2 プロファイルキーワード

プロファイルキーワード	インストール方法				
	スタンドアロンシステム (ネットワークに接続されていない)	スタンドアロンシステム (ネットワークに接続されている) または サーバー	OS サーバー	アップグレード	ディスク容量の再配置を使用するアップグレード
archive_location	X	X			
backup_media					X
boot_device	X	X	X		
client_arch			X		
client_root			X		
client_swap			X		
cluster (ソフトウェアグループを追加する場合)	X	X	X		
cluster (クラスタを追加または削除する場合)	X	X	X	X	X
dontuse	X	X	X		
filesystem (リモートファイルシステムをマウントする場合)		X	X		
filesystem (ローカルファイルシステムを作成する場合)	X	X	X		
geo	X	X	X	X	X
install_type	X	X	X	X	X
isa_bits	X	X	X	X	X
layout_constraint					X
locale	X	X	X	X	X
num_clients			X		
package	X	X	X	X	X
partitioning	X	X	X		
root_device	X	X	X	X	X
system_type	X	X	X		
usedisk	X	X	X		

プロファイルキーワードの説明と例

archive_location プロファイルキーワード

`archive_location retrieval_type location`

`retrieval_type` と `location` の値は、フラッシュアーカイブの格納場所によって異なります。`retrieval_type` と `location` に使用できる値や、`archive_location` キーワードの使用例については、次の各項を参照してください。

- 204 ページの「ネットワークファイルシステム (NFS) サーバー」
- 204 ページの「HTTP サーバー」
- 206 ページの「FTP サーバー」
- 207 ページの「ローカルテープ」
- 207 ページの「ローカルデバイス」
- 208 ページの「ローカルファイル」

ネットワークファイルシステム (NFS) サーバー

アーカイブがネットワークファイルシステム (NFS) サーバーに格納されている場合は、`archive_location` プロファイルキーワードに次の構文を使用します。

```
archive_location nfs server_name:/path/filename retry n
```

<code>server_name</code>	アーカイブを格納するサーバーの名前です。
<code>path</code>	指定したサーバーから取得するアーカイブの場所です。このパスに <code>\$HOST</code> を含めると、フラッシュインストールユーティリティは <code>\$HOST</code> をインストール先のクローンシステムの名前に置き換えます。
<code>filename</code>	フラッシュアーカイブファイルの名前です。
<code>retry n</code>	省略可能なキーワードです。 <code>n</code> は、フラッシュユーティリティがアーカイブのマウントを試みる最大回数です。

例:

```
archive_location nfs golden:/archives/usrarchive
```

```
archive_location nfs://golden/archives/usrarchive
```

HTTP サーバー

アーカイブが HTTP サーバーに格納されている場合は、`archive_location` プロファイルキーワードに次の構文を使用します。

```
archive_location http://server_name:port path/filename optional_keywords
```

<i>server_name</i>	アーカイブを格納するサーバーの名前です。 <i>server_name</i> は、ポート番号でも、実行時に決定されるポート番号を持つ TCP サービスの名前でもかまいません。
<i>port</i>	オプションのポートです。ポートを指定しない場合、フラッシュインストールユーティリティはデフォルトの HTTP ポート番号 80 を使用します。
<i>path</i>	指定したサーバーから取得するアーカイブの場所です。このパスに \$HOST を含めると、フラッシュインストールユーティリティは \$HOST をインストール先のクローンシステムの名前に置き換えます。
<i>filename</i>	フラッシュアーカイブファイルの名前です。
<i>optional_keywords</i>	フラッシュアーカイブを HTTP サーバーから取得するときに指定できるオプションのキーワードです。

表 28-3 *optional_keywords* で使用可能なオプションのキーワード

キーワード	値の定義
<code>auth basic user_name password</code>	<p>アーカイブがパスワード保護された HTTP サーバーに格納されている場合は、その HTTP サーバーへのアクセスに必要なユーザー名とパスワードをプロファイルに含める必要があります。</p> <p>注 - カスタム JumpStart で使用しようとするプロファイルにおいてこの認証方法を使用することは、リスクを伴います。これは、承認されていないユーザーが、パスワードが入ったプロファイルにアクセスできる可能性があるためです。</p>
<code>timeout min</code>	<p><code>timeout</code> キーワードには、HTTP サーバーからのデータ受信を待機する最長の時間を分単位で指定できます。この時間に達すると、接続が切断されて、再接続が行われ、タイムアウトが発生した地点から再開されます。<code>timeout</code> 値として 0 (ゼロ) を指定すると、何も起きないため再接続されません。</p> <p>タイムアウトによる再接続が発生すると、フラッシュインストールユーティリティはアーカイブを取得した最後の位置からインストールの再開を試みます。この位置でのインストールの再開が不可能な場合、アーカイブの初めから再度データの取得が行われ、タイムアウト前に取得されたデータは破棄されます。</p>
<code>proxy host:port</code>	<p><code>proxy</code> キーワードを使用して、プロキシホストとプロキシポートを指定できます。プロキシホストを使用すると、ファイアウォール越しにフラッシュアーカイブを取得できます。<code>proxy</code> キーワードを指定する場合は、プロキシポートを指定する必要があります。</p>

例:

```
archive_location http://silver/archives/usrarchive.flar timeout 5
```

`auth basic user_name password` キーワードの例:

```
archive_location http://silver/archives/usrarchive.flar timeout 5 user1 secret
```

FTP サーバー

アーカイブが FTP サーバーに格納されている場合は、`archive_location` プロファイルキーワードに次の構文を使用します。

```
archive_location ftp://user_name:password@server_name:port path/filename optional_keywords
```

<i>user_name:password</i>	プロファイルファイルにおいて、FTP サーバーにアクセスするために必要なユーザー名とパスワードです。
<i>server_name</i>	アーカイブを格納するサーバーの名前です。 <i>server_name</i> は、ポート番号でも、実行時に決定されるポート番号を持つ TCP サービスの名前でもかまいません。
<i>port</i>	オプションのポートです。ポートを指定しない場合、フラッシュインストールユーティリティはデフォルトの FTP ポート番号 21 を使用します。
<i>path</i>	指定したサーバーから取得するアーカイブの場所です。このパスに \$HOST を含めると、フラッシュインストールユーティリティは \$HOST をインストール先のクローンシステムの名前に置き換えます。
<i>filename</i>	フラッシュアーカイブファイルの名前です。
<i>optional_keywords</i>	フラッシュアーカイブを FTP サーバーから取得するときに指定できるオプションのキーワードです。

表 28-4 `archive_location ftp` で使用可能なオプションのキーワード

キーワード	値の定義
<i>timeout min</i>	<p><code>timeout</code> キーワードには、FTP サーバーからのデータ受信を待機する最長の時間を分単位で指定できます。この時間に達すると、接続が切断されて、再接続が行われ、タイムアウトが発生した地点から再開されます。<code>timeout</code> 値として 0 (ゼロ) を指定すると、何も起きないため再接続されません。</p> <p>タイムアウトによる再接続が発生すると、フラッシュインストールユーティリティはアーカイブを取得した最後の位置からインストールの再開を試みます。この位置でのインストールの再開が不可能な場合、アーカイブの初めから再度データの取得が行われ、タイムアウト前に取得されたデータは破棄されます。</p>
<i>proxy host:port</i>	<p><code>proxy</code> キーワードを使用して、プロキシホストとプロキシポートを指定できます。プロキシホストを使用すると、ファイアウォール越しにフラッシュアーカイブを取得できます。<code>proxy</code> キーワードを指定する場合は、プロキシポートを指定する必要があります。</p>

例:

```
archive_location ftp://user1:secret@silver/archives/usrarchive.flar timeout 5
```

ローカルテープ

アーカイブがテープに格納されている場合は、`archive_location` プロファイルキーワードに次の構文を使用します。

```
archive_location local_tape device position
```

<i>device</i>	フラッシュアーカイブを格納したテープドライブの名前です。デバイス名が正規のパスである場合は、フラッシュインストールユーティリティはデバイスノードへのパスからアーカイブを取得します。正規のパスでないデバイス名を指定した場合は、フラッシュインストールユーティリティはパスに <code>/dev/rmt/</code> を加えます。
<i>position</i>	アーカイブを保存したテープドライブ上の位置を示します。位置が指定されないと、フラッシュインストールユーティリティはテープドライブの現在の位置からアーカイブを取得します。 <i>position</i> を指定することにより、テープドライブ上でアーカイブの前に開始スクリプトまたは <code>sysidcfg</code> ファイルを配置できます。

例:

```
archive_location local_tape /dev/rmt/0n 5
```

```
archive_location local_tape 0n 5
```

ローカルデバイス

ファイルシステム指向のランダムアクセスデバイス (フロッピーディスクや CD-ROM など) にフラッシュアーカイブを格納した場合は、ローカルデバイスからフラッシュアーカイブを取得できます。`archive_location` プロファイルキーワードには次の構文を使用します。

注 - ローカルテープ用の構文を使用すると、ストリーム指向のデバイス (テープなど) からアーカイブを取得できます。

```
archive_location local_device device path/filename file_system_type
```

<i>device</i>	フラッシュアーカイブを格納したドライブの名前です。デバイス名が正規のパスである場合は、デバイスは直接マウントされます。正規のパスでないデバイス名を指定すると、フラッシュインストールユーティリティはパスに <code>/dev/dsk/</code> を加えます。
<i>path</i>	フラッシュアーカイブへのパスです (指定したデバイス上のファイルシステムのルートからの相対的なパス)。このパスに <code>\$HOST</code> を含めると、フラッシュインストールユーティリティは <code>\$HOST</code> をインストール先のクローンシステムの名前に置き換えます。

filename フラッシュアーカイブファイルの名前です。

file_system_type デバイス上のファイルシステムのタイプを指定します。ファイルシステムのタイプを指定しない場合、フラッシュインストールユーティリティは、UFS ファイルシステムのマウントを試みます。UFS のマウントに失敗すると、フラッシュインストールユーティリティは HSFS ファイルシステムのマウントを試みます。

例:

UFS ファイルシステムとしてフォーマットされているローカルハードディスクからアーカイブを取得するには、次のコマンドを使用します。

```
archive_location local_device c0t0d0s0 /archives/$HOST
```

HSFS ファイルシステムを持つローカル CD-ROM からアーカイブを取得するには、次のコマンドを使用します。

```
archive_location local_device c0t0d0s0 /archives/usrarchive
```

ローカルファイル

クローンシステムをブートしたミニルートに格納したローカルファイルのアーカイブを取得できます。カスタム JumpStart インストールを実施する時に、CD-ROM または NFS ベースのミニルートからシステムをブートします。このミニルートからインストールソフトウェアがロードされ、実行されます。したがって、CD-ROM または NFS ベースのミニルートに格納したフラッシュアーカイブは、ローカルファイルとしてアクセスできます。archive_location プロファイルキーワードには次の構文を使用します。

```
archive_location local_file path/filename
```

path アーカイブの位置です。このパスは、システムを Solaris 9 INSTALLATION CD または Solaris 9 DVD からブートしている間、システムにローカルファイルとしてアクセスできるものでなければなりません。Solaris 9 INSTALLATION CD または Solaris 9 DVD からブートしている間は、システムは /net にアクセスできません。

filename フラッシュアーカイブファイルの名前です。

例:

```
archive_location local_file /archives/usrarchive
```

backup_media プロファイルキーワード

```
backup_media type path
```

注 - backup_media は、ディスク容量の再配置が必要なアップグレードオプションだけで使用できません。

backup_media は、ディスク容量不足のためにアップグレード中にディスク容量の再配置が必要なファイルシステムのバックアップをとるために使用するメディアを定義します。バックアップ用に複数のテープまたはフロッピーディスクが必要な場合は、アップグレード中にテープまたはフロッピーディスクの挿入を求めるプロンプトが表示されます。

有効な <i>type</i> 値	有効な <i>path</i> 値	仕様
local_tape	/dev/rmt/ <i>n</i>	アップグレードされるシステムのローカルテープドライブを指定する。 <i>path</i> は、テープドライブのキャラクタ型 (raw) デバイスパスである必要がある。 <i>n</i> はテープドライブの番号
local_diskette	/dev/rdisketten	アップグレードされるシステムのローカルフロッピーディスクドライブを指定する。 <i>path</i> は、フロッピーディスクドライブのキャラクタ型 (raw) デバイスパスである必要がある。 <i>n</i> はフロッピーディスクドライブの番号 バックアップに使用するフロッピーディスクはフォーマットされていない
local_filesystem	/dev/dsk/cwtxdysz /file_system	アップグレードされるシステムのローカルファイルシステムを指定する。アップグレードで変更されるローカルファイルシステムは指定できない。 <i>path</i> は、ディスクスライスのブロックデバイスパスでもかまわない。たとえば、/dev/dsk/c wtx dysz 内の <i>tx</i> は必須ではない。あるいは、 <i>path</i> は /etc/vfstab ファイルによってマウントされたファイルシステムへの絶対パスでもかまわない
remote_filesystem	host: /file_system	リモートシステムの NFS ファイルシステムを指定する。 <i>path</i> は、リモートシステム (<i>host</i>) の名前または IP アドレスと、NFS ファイルシステム (<i>file_system</i>) への絶対パスを含まなければならない。NFS ファイルシステムは、読み取り権と書き込み権を持っている必要がある

有効な <i>type</i> 値	有効な <i>path</i> 値	仕様
remote_system	<i>user@host:/directory</i>	リモートシェル (rsh) で到達できるリモートシステム上のディレクトリを指定する。アップグレードされるシステムは、リモートシステムの <i>.rhosts</i> ファイル経由で、リモートシステムにアクセスできなければならない。 <i>path</i> は、リモートシステム (<i>host</i>) の名前と、そのディレクトリ (<i>directory</i>) への絶対パスを含まなければならない。ユーザーログイン (<i>user</i>) を指定しないと、スーパーユーザーとしてログインされる

例:

```

backup_media local_tape /dev/rmt/0

backup_media local_diskette /dev/rdiskette1

backup_media local_filesystem /dev/dsk/c0t3d0s4

backup_media local_filesystem /export

backup_media remote_filesystem system1:/export/temp

backup_media remote_system user1@system1:/export/temp

```

boot_device プロファイルキーワード

`boot_device device eeprom`

`boot_device` には、JumpStart がルート (/) ファイルシステムをインストールするデバイスと、システムのブートデバイスを指定します。

`boot_device` キーワードをプロファイルに指定しない場合、インストール中にデフォルトで次の `boot_device` キーワードが指定されます。 `boot_device any update`

device - 以下の値のいずれかを使用して、ブートデバイスにするデバイスを指定します。

- *cwtxdysz* または *cxdysz* - JumpStart プログラムがルート (/) ファイルシステムを格納するディスクスライス。たとえば、*c0t0d0s0*。
- *existing* - JumpStart プログラムは、システムの既存のブートデバイスにルート (/) ファイルシステムを格納します。
- *any* - JumpStart プログラムは、ルート (/) ファイルシステムを配置する場所を選択します。システムの既存のブートデバイスを使用する場合がありますが、必要であれば異なるブートデバイスを選択します。

eeprom - システムの EEPROM を変更または保存する場合に選択します。

システムの EEPROM を、指定したブートデバイスに変更または保存する場合に選択します。

- `update` - JumpStart プログラムは、インストールされるシステムが自動的に指定したブートデバイスからブートするように、システムの EEPROM をそのブートデバイスに変更します。
- `preserve` - システムの EEPROM 中のブートデバイス値は変更されません。システムの EEPROM を変更しないで新しいブートデバイスを指定した場合は、システムが新しいブートデバイスから自動的にブートするように、システムの EEPROM を手作業で変更する必要があります。

注 - `eprom` 値を使用すると、システムの現在のブートデバイスを変更した場合にシステムの EEPROM を更新できます。これにより、システムは新しいブートデバイスから自動的にブートできます。

例:

```
boot_device c0t0d0s2 update
```

注 - `boot_device` は、ルートファイルシステムを指定する `filesys` キーワードと `root_device` キーワード (指定した場合) に一致する必要があります。

client_arch プロファイルキーワード

```
client_arch karch_value ...
```

`client_arch` は、OS サーバーが、それ自体が使用するものとは異なるプラットフォームグループをサポートすることを定義します。`client_arch` を指定しない場合、OS サーバーを使用するどのディスクレスクライアントも、サーバーと同じプラットフォームグループでなくてはなりません。OS サーバーがサポートしてほしいプラットフォームグループごとに指定する必要があります。

`karch_value` の有効な値は、`sun4m` と `sun4u` です。プラットフォーム名と各種システムについての詳細は、『Solaris 9 Sun ハードウェアマニュアル』を参照してください。

注 - `client_arch` は、`system_type` に `server` を指定したときだけ使用できます。

client_root プロファイルキーワード

```
client_root root_size
```

`client_root` は、各クライアント用に割り当てるルート領域の大きさ (`root_size`、M バイト単位) を定義します。サーバーのプロファイルに `client_root` を指定しないと、インストールソフトウェアは1つのクライアント当たり 15M バイトのルート領域を割り当てます。このクライアント用のルート領域の大きさは、`num_clients` キーワードを組み合わせ、`/export/root` ファイルシステム用に確保する領域の大きさを決定するときに使用されます。

注 - `client_root` は、`system_type` に `server` を指定したときだけ使用できません。

client_swap プロファイルキーワード

`client_swap swap_size`

プロファイル内の `client_swap` は、各ディスクレスクライアントに割り当てるスワップ領域の大きさ (`swap_size`、M バイト単位) を定義します。`boot_device` キーワードをプロファイルに指定しない場合、インストール中にデフォルトで次の `boot_device` キーワードが指定されます。

例:

```
client_swap 64
```

この例は、各ディスクレスクライアントが 64M バイトのスワップ領域を持つことを定義します。

注 - `client_swap` は、`system_type` に `server` を指定したときだけ使用できません。

swap のサイズを決定する方法

プロファイルが `swap` のサイズを指定していない場合、JumpStart プログラムはシステムの物理メモリーに基づいてスワップ空間のサイズを決定します。表 28-5 に、カスタム JumpStart インストール中に `swap` のサイズがどのように決定されるかを示します。

表 28-5 `swap` のサイズの決定

物理メモリー (単位: M バイト)	スワップ領域 (単位: M バイト)
16-64	32
64-128	64

表 28-5 swap のサイズの決定 (続き)

物理メモリー (単位: M バイト)	スワップ領域 (単位: M バイト)
128-512	128
512 を超える場合	256

JumpStart プログラムは swap のサイズが swap が作成されるディスクの 20% を超えないようにします。ただし、他のファイルシステムを配置した後に空き領域が残っている場合を除きます。空き領域が残っている場合、JumpStart プログラムは空き領域を swap に割り当てて、可能であれば、表 28-5 に示す量を割り当てます。

注 - 物理メモリーとスワップ領域の合計は、32M バイト以上必要です。

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)

`cluster group_name`

`cluster` は、どのソフトウェアグループをシステムに追加するかを指定します。各ソフトウェアグループの `group_name` 名は次のとおりです。

ソフトウェアグループ	group_name
コアシステムサポート	SUNWCreq
エンドユーザーシステムサポート	SUNWCuser
開発者システムサポート	SUNWCprog
全体ディストリビューション	SUNWCall
全体ディストリビューションと OEM サポート	SUNWCxall

1 つのプロファイルに指定できるソフトウェアグループの数は 1 つだけです。ソフトウェアグループは、ほかの `cluster` エントリおよび `package` エントリよりも前に指定する必要があります。プロファイル内の `cluster` でソフトウェアグループを指定しない場合、デフォルトにより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 (SUNWCuser) がシステムにインストールされます。

cluster プロファイルキーワード (クラスタの追加または削除)

`cluster cluster_name add_delete_switch`

注 - `cluster` (クラスタの追加または削除) は、初期インストールオプションとアップグレードオプションの両方で使用できます。

`cluster` は、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クラスタを追加または削除するかを指定します。

`cluster_name` は `SUNWCname` 形式で指定する必要があります。インストールが終了したシステムで `Admintool` を起動し、「ブラウザ」メニューから「ソフトウェア」を選択すると、クラスタの詳細情報とクラスタ名を表示できます。

`add_delete_switch` は、オプション `add` または `delete` を示します。このオプションを使用すると、指定したクラスタを追加または削除できます。`add_delete_switch` を指定しないと、デフォルトによって `add` が使用されます。

次に、アップグレード時に `cluster` (クラスタの追加または削除) を使用した場合の動作を示します。

- すでにシステムにあるすべてのクラスタが自動的にアップグレードされます。
- `cluster_name add` を指定したが、`cluster_name` がシステムにインストールされていなかった場合、そのクラスタがインストールされます。
- `cluster_name delete` を指定したが、`cluster_name` がシステムにインストールされていた場合、アップグレードが開始される前にそのパッケージは削除されます。

dontuse プロファイルキーワード

`dontuse disk_name ...`

`partitioning default` が指定されると、デフォルトではシステム上のすべての使用可能ディスクが使用されます。`dontuse` は、`JumpStart` プログラムに使用させないディスク (1 つ以上) を指定するために使用します。`disk_name` は `cxt ydz` 形式 (たとえば、`c0t0d0`) で指定する必要があります。

注 - 1 つのプロファイルで、`dontuse` キーワードと `usedisk` キーワードを同時に指定することはできません。

filesystem プロファイルキーワード (リモートファイルシステムのマウント)

`filesystem server:path server_address mount_pt_name [mount_options]`

これらの値を指定して `filesystem` を使用すると、`JumpStart` プログラムはインストールされているシステムがそのブート時に自動的にリモートファイルシステムをマウントするように設定します。`filesystem` は 2 回以上指定できます。

例:

```
filesystem sherlock:/export/home/user2 - /home
```

server-リモートファイルシステムが存在するサーバー名 (後ろにコロンを付けます)

path- リモートファイルシステムのマウントポイント名 (例: /usr、 /export/home など)

server_address - *server:path* で指定するサーバーの IP アドレス。ネットワーク上で実行されているネームサービスがない場合、この値 *server_address* を使用して、サーバーのホスト名とIPアドレスを登録している /etc/hosts ファイルを生成できます。サーバーの IP アドレスを指定しない場合は、マイナス記号 (-) を指定してください。たとえば、ネットワーク上で実行中のネームサービスがある場合、サーバーの IP アドレスを指定する必要がありません。

mount_pt_name - リモートファイルシステムをマウントするマウントポイント名

mount_options - 1 つ以上のマウントオプション。これは、mount (1M) コマンドの -o オプションと同じです。このマウントオプションは、指定された *mount_pt_name* の /etc/vfstab エントリに追加されます。

注 - 複数のマウントオプションを指定する場合は、マウントオプションはスペースではなくコンマで区切ってください。例: ro, quota

filesystem プロファイルキーワード (ローカルファイルシステムの作成)

```
filesystem slice size [file_system optional_parameters]
```

これらの値を指定して filesystem を使用すると、JumpStart プログラムによりインストール時にローカルファイルシステムが作成されます。filesystem は 2 回以上指定できません。

slice - 次の中から 1 つを指定します。

- any - JumpStart プロファイルは、ファイルシステムを任意のディスクに配置します。

注 - size が existing、all、free、start:size、または ignore の場合は、any は指定できません。

- cw txdy sz - JumpStart プログラムがファイルシステムを配置するディスクライイス。たとえば、c0t0d0s0 または c0d0s0。

- `rootdisk.sn` - システムのルートディスク値が含まれる変数。224 ページの「システムのルートディスクを決定する方法」に示されているように、この値は `JumpStart` プログラムが決定します。拡張子 `sn` は、ディスク上の特定のスライスを示します。

`size` - 次の中から 1 つを使用します。

- `num` - ファイルシステムのサイズを `num` (M バイト単位) で設定します。
- `existing` - 既存のファイルシステムの現在のサイズを使用します。

注 - `existing` の値を使用すると、別の `mount_pt_name` として `file_system` を指定することによって、既存のスライス名を変更できます。

- `auto` - 選択したソフトウェアに応じて、ファイルシステムのサイズを自動的に決定します。
- `all` - 指定した `slice` は、そのファイルシステム用にディスク全体を使用します。この値を指定すると、指定したディスク上にほかのファイルシステムは存在できません。
- `free` - ディスク上の残りの未使用領域をファイルシステム用に使用します。

注 - `filesys` の値として `free` を使用する場合は、`filesys` はプロファイルの最後のエントリにする必要があります。

- `start:size` - ファイルシステムを明示的にパーティションに分割します。`start` はスライスが始まるシリンダであり、`size` はそのスライスのシリンダ数です。

`file_system` - `slice` に `any` または `cwtxdysz` を指定しているときに、このオプション値を使用できます。この値を指定しないと `unnamed` がデフォルトで設定されますが、この場合 `optional_parameters` 値を指定できません。次のいずれか 1 つを指定してください。

- `mount_pt_name` - ファイルシステムのマウントポイント名 (例: `/var`)。
- `swap` - 指定した `slice` を `swap` として使用します。
- `overlap` - 指定した `slice` をディスク領域 (VTOC 値は `V_BACKUP`) を表すものとして定義します。デフォルトでは、スライス 2 はディスク全体を表すオーバーラップスライスです。

注 - `size` に `existing`、`all`、または `start:size` を指定した場合だけ `overlap` を指定できます。

- `unnamed` - 指定した `slice` は raw スライスと定義されるので、`slice` にはマウントポイント名がありません。`file_system` を指定しないと、デフォルトで `unnamed` が設定されます。
- `ignore` - 指定した `slice` を使用しないか、JumpStart プログラムで認識しません。このオプションは、インストール時にディスク上の特定のファイルシステムを無視させるために使用できます。JumpStart プログラムは、同じディスク上に同じ名前での新しいファイルシステムを作成します。`ignore` は、`partitioning existing` を指定したときだけ使用できます。

`optional_parameters` - 次のいずれか 1 つを使用します。

- `preserve` - 指定した `slice` 上のファイルシステムを保存します。

注 - `size` に `existing`、`slice` に `cwtxdysz` を指定した場合だけ `preserve` を指定できます。

- `mount_options` - 1 つ以上のマウントオプション (`mount (1M)` コマンドの `-o` オプションと同じ)。これらのマウントオプションは、指定された `mount_pt_name` の `/etc/vfstab` エントリに追加されます。

注 - 複数のマウントオプションを指定する場合は、マウントオプションはスペースではなくコンマで区切ってください。例: `ro,quota`

geo プロファイルキーワード

`geo locale`

注 - `geo` は、初期インストールとアップグレードオプションの両方で使用できます。

`geo` は、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。`locale` に指定できる値は、次のとおりです。

値	説明
<code>N_Africa</code>	北アフリカ。エジプトを含む
<code>C_America</code>	中央アメリカ。コスタリカ、エルサルバドル、グアテマラ、メキシコ、ニカラグア、パナマを含む

値	説明
N_America	北アメリカ。カナダ、アメリカ合衆国を含む
S_America	南アメリカ。アルゼンチン、ボリビア、ブラジル、チリ、コロンビア、エクアドル、パラグアイ、ペルー、ウルグアイ、ベネズエラを含む
Asia	アジア。日本、韓国、中華人民共和国、台湾、タイを含む
Ausi	オーストラリア。オーストラリア、ニュージーランドを含む
C_Europe	中央ヨーロッパ。オーストリア、チェコ、ドイツ、ハンガリー、ポーランド、スロヴァキア、スイスを含む
E_Europe	東ヨーロッパ。アルバニア、ボスニア、ブルガリア、クロアチア、エストニア、ラトビア、リトアニア、マケドニア、ルーマニア、ロシア、セルビア、スロヴェニア、トルコを含む
N_Europe	北ヨーロッパ。デンマーク、フィンランド、アイスランド、ノルウェー、スウェーデンを含む
S_Europe	南ヨーロッパ。ギリシャ、イタリア、ポルトガル、スペインを含む
W_Europe	西ヨーロッパ。ベルギー、フランス、イギリス、アイルランド、オランダを含む
M_East	中近東。イスラエルを含む

上記の各地域ロケールを構成するコンポーネントロケール値の完全なリストは、第 40 章に記載されています。

注 - システムに追加する必要がある各ロケールごとに、geo キーワードを指定します。

install_type プロファイルキーワード

`install_type initial_upgrade_flash_swith`

`install_type` は、システムにおいて、(既存の Solaris オペレーティング環境を) 消去して、新しい Solaris オペレーティング環境をインストールするか、既存の Solaris オペレーティング環境をアップグレードするか、あるいは、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。

注 - `install_type` は必須であり、各プロファイル内で最初のプロファイルキーワードとして指定する必要があります。

initial_upgrade_flash_switch には、次のオプションのうちの 1 つを使用する必要があります。

- *initial_install* – Solaris オペレーティング環境を初期インストールすることを指定します。
- *upgrade* – Solaris オペレーティング環境をアップグレードすることを指定します。
- *flash_install* – フラッシュアーカイブをインストールすることを指定します。

注 – プロファイルキーワードの中には、*initial_install* オプションでしか使用できないものがあります。*upgrade* オプションでしか使用できないものもあります。また、*flash_install* オプションでしか使用できないものがあります。

isa_bits プロファイルキーワード

isa_bits bit_switch

isa_bits は、64 ビットまたは 32 ビットの Solaris 9 パッケージをインストールするかどうかを指定します。

bit_switch は、オプション 64 または 32 を表します。これは、64 ビットまたは 32 ビットのどちらの Solaris 9 パッケージをインストールするかを指定するために使用します。このキーワードをプロファイルに設定しないと、JumpStart プログラムによって、次のものがインストールされます。

- UltraSPARC™ システムの場合は 64 ビットパッケージ
- それ以外のシステムの場合は 32 ビットパッケージ

注 – *isa_bits* キーワードを使用する場合は、Solaris 9 SOFTWARE 1 of 2 CD または Solaris 9 DVD 上の *solaris_9/Misc/jumpstart_sample* ディレクトリの最新の check スクリプトを使用する必要があります。

layout_constraint プロファイルキーワード

layout_constraint slice constraint [minimum_size]

注 – *layout_constraint* は、ディスク容量の再配置が必要なアップグレードオプションだけで使用できます。

layout_constraint は、ファイルシステムがディスク容量不足のためにアップグレード中にディスク容量を再配置する必要がある場合に、制約付き自動配置がファイルシステムで行われることを示します。

layout_constraint キーワードを指定しないと、次のように配置されます。

- アップグレード用により多くの容量を必要とするファイルシステムは、changeable とマークされます。
- より多くの容量を必要とするファイルシステムと同じディスク上にあるファイルシステム (/etc/vfstab ファイルでマウントされる) も changeable とマークされま
- 残りのファイルシステムは fixed とマークされます。これは、自動配置がこれらのファイルシステムを変更できないためです。

1 つ以上の layout_constraint キーワードを指定すると、次のように配置されま

- アップグレード用により多くの容量を必要とするファイルシステムは、changeable とマークされます。
- layout_constraint キーワードを指定したファイルシステムは、指定した制約がマークされます。
- 残りのファイルシステムは、fixed とマークされます。

アップグレードにより多くの容量を必要とするファイルシステムの制約は変更できませんが (changeable とマークされなければならない)、このようなファイルシステムに layout_constraint キーワードを使用すれば、その minimum_size 値を変更できま

注 - 自動配置がディスク容量の再配置を行う際には、より多くのファイルシステム、特にアップグレード用により多くの容量を必要とするファイルシステムと同じディスク上にあるファイルシステムを、changeable または movable であると選択します。

slice - これは、制約を指定するファイルシステムのディスクスライスです。システムのディスクスライスは cwtxdysz 形式で指定する必要があります。

constraint - 指定したファイルシステムに対して、次のいずれか 1 つの制約を選択しま

- changeable - 自動配置はファイルシステムをほかの場所に移動して、そのサイズを変更できます。この制約は、/etc/vfstab ファイルによってマウントされたファイルシステムにしか指定できません。minimum_size 値を指定すれば、ファイルシステムのサイズを変更できます。

ファイルシステムを changeable とマークして、minimum_size 値を指定しないと、そのファイルシステムの最小サイズは、必要な最小サイズより 10% 大きな値に設定されます。たとえば、ファイルシステムの最小サイズが 100M バイトの場合、変更されるサイズは 110M バイトになります。minimum_size を指定した場合、残りの空き領域 (元のサイズから最小サイズを引いたもの) はほかのファイルシステム用に使用されます。

- movable - 自動配置はファイルシステムを (同じディスクまたは異なるディスク上の) ほかのスライスに移動できますが、サイズは変更しません。

- `available` - 自動配置は、ファイルシステムのすべての領域を使用して領域を割り当て直します。ファイルシステムのすべてのデータは失われます。この制約は、`/etc/vfstab` ファイルでマウントされないファイルシステムだけに指定できます。
- `collapse` - 自動配置は、指定したファイルシステムをその親ファイルシステムに移動して閉じこめます。このオプションは、アップグレードの一部としてシステム上のファイルシステム数を減らすために使用できます。たとえば、システムにファイルシステム `/usr` と `/usr/share` が存在する場合、`/usr/share` ファイルシステムを閉じ込めると、このファイルシステムは `/usr` (その親) に移動します。この制約は、`/etc/vfstab` ファイルでマウントされるファイルシステムにしか指定できません。

`minimum_size` - この値は、自動配置がディスク容量を再配置するときに、ファイルシステムに割り当てる最小サイズを指定します (基本的にファイルシステムのサイズを変更する)。まだ割り当てられていない領域が追加される場合、ファイルシステムのサイズは最終的にこの指定した値より大きくなる可能性があります。しかし、指定される値よりサイズが小さくなることはありません。`minimum_size` 値は省略可能です。このオプション値を使用できるのは、ファイルシステムを `changeable` とマークした場合だけです。最小サイズは、ファイルシステムの既存の内容に必要なサイズより小さい値には設定できません。

例:

```
layout_constraint c0t3d0s1 changeable 200

layout_constraint c0t3d0s4 movable

layout_constraint c0t3d1s3 available

layout_constraint c0t2d0s1 collapse
```

locale プロファイルキーワード

locale *locale_name*

注 - `locale` は、初期インストールとアップグレードオプションの両方で使用できます。

`locale` は、指定した *locale_name* に対して、どのロケールパッケージをインストール (アップグレードの場合は追加) するかを指定します。*locale_name* 値は、`$LANG` 環境変数で使用されるのと同じです。有効なロケールの値については、第 40 章を参照してください。

`local` キーワードを使用する場合は、次の点を考慮してください。

- デフォルトロケールを事前設定している場合は、そのロケールは自動的にインストールされます。`English` 言語パッケージはデフォルトでインストールされます。

- locale キーワードは、システムに追加するロケールごとに指定できます。

num_clients プロファイルキーワード

num_clients *client_num*

サーバーがインストールされているときには、各ディスクレスクライアントのルート (/) と swap ファイルシステムにディスク空間が割り当てられます。num_clients は、サーバーがサポートするディスクレスクライアント数 (*client_num*) を定義します。num_clients を指定しないと、デフォルトで5つのディスクレスクライアントが割り当てられます。

注 - num_clients は、system_type が server として指定されているときだけ使用できます。

package プロファイルキーワード

package *package_name* [*add_delete_switch*]

注 - package は、初期インストールとアップグレードオプションの両方で使用できます。

package は、システムにインストールするソフトウェアグループにパッケージを追加または削除するかを指定します。

package_name は、SUNWname 形式で指定する必要があります。パッケージとその名前の詳細を表示するには、インストール済みシステムで pkginfo -l コマンドを使用します。

add_delete_switch は、add または delete オプションを表します。このオプションは、指定のパッケージを追加または削除するかを指定します。add_delete_switch を指定しないと、デフォルトによって add が使用されます。

注 - いくつかのパッケージは必須であり、削除できないものもあります。各国語対応パッケージを個々に追加または削除するとき、package プロファイルキーワードは使用できません。各国語対応パッケージを追加するには、locale プロファイルキーワードを使用します。

アップグレードに package を使用すると、次の処理が行われます。

- すでにシステム上にあるすべてのパッケージが自動的にアップグレードされます。

- `package_name add` を指定したが、`package_name` がシステムにインストールされていなかった場合は、そのパッケージがインストールされます。
- `package_name delete` を指定したが、`package_name` がシステムにインストールされていた場合、アップグレードが開始される前にそのパッケージは削除されます。
- `package_name delete` を指定したが、`package_name` がシステムにインストールされていない場合、インストールするように指定したクラスタの一部にそのパッケージが含まれていると、パッケージはインストールされません。

partitioning プロファイルキーワード

`partitioning type`

`partitioning` は、インストール時にファイルシステム用にディスクをスライスに分割する方法を定義します。

`type` – 次の中から 1 つを指定します。

- `default` – JumpStart プログラムはディスクを選択して、指定したソフトウェアをインストールするファイルシステムを作成します。ただし、`filesys` キーワードで指定したファイルシステムを除きます。`rootdisk` が最初に選択され、指定したソフトウェアが `rootdisk` に収まらない場合は、さらに別のディスクが使用されます。
- `existing` – JumpStart プログラムは、システムのディスク上にある既存のファイルシステムを使用します。`/`、`/usr`、`/usr/openwin`、`/opt`、`/var` を除く、すべてのファイルシステムが保存されます。JumpStart プログラムは、ファイルシステムのスーパーブロックにある最後のマウントポイントフィールドを使用して、スライスがどのファイルシステムのマウントポイントを表しているかを判断します。

注 – `filesys` プロファイルキーワードと `partitioning existing` を組み合わせる場合、`size` は `existing` である必要があります。

- `explicit` – JumpStart プログラムは、ディスクを使用し、`filesys` キーワードで指定されるファイルシステムを作成します。`filesys` キーワードでルート (`/`) ファイルシステムだけを指定した場合、すべての Solaris ソフトウェアがルートファイルシステムにインストールされます。

注 – `explicit` プロファイル値を使用するときには、`filesys` プロファイルキーワードを使用して、使用するディスクと作成するファイルシステムを指定してください。

プロファイルで `partitioning` を指定しないと、デフォルトで `default` タイプのパーティションが使用されます。

root_device プロファイルキーワード

root_device slice

注 - root_device は、初期インストールとアップグレードオプションの両方で使用できます。

root_device は、システムのルートディスクを指定します。詳細は、224 ページの「システムのルートディスクを決定する方法」を参照してください。

システムをアップグレードする場合、root_device は指定されるルート (/) ファイルシステムおよびその /etc/vfstab ファイルでマウントされるファイルシステムがアップグレードされることを示します。システム上で複数のルートファイルシステムがアップグレードできる場合は、root_device を指定する必要があります。slice は cwtxdysz 形式で指定する必要があります。

例:

```
root_device c0t0d0s2
```

root_device キーワードを使用する場合は、次の点を考慮してください。

- 1 つだけのディスクを持つシステムで root_device を指定する場合、root_device とディスクが一致する必要があります。また、ルートファイルシステムを指定する任意の filesys キーワードは、root_device と一致する必要があります。
- ミラーをアップグレードする場合、root_device に指定する値はミラーの一方である必要があります。他方のミラーは自動的にアップグレードされます。

システムのルートディスクを決定する方法

システムのルートディスクは、ルート (/) ファイルシステムを含むシステム上のディスクです。プロファイル内では、JumpStart プログラムがシステムのルートディスクを設定するディスク名の代わりに、この rootdisk 変数を使用できます。表 28-6 に、JumpStart プログラムがインストール用にシステムのルートディスクを決定する方法を説明しています。

注 - システムのルートディスクサイズが確認されるのは、初期インストール時だけです。アップグレードの場合、システムのルートディスクは変更できません。

表 28-6 JumpStart がシステムのルートディスクを決定する方法 (初期インストールのみ)

手順	動作
1	プロファイル内で <code>root_device</code> キーワードが指定されている場合、JumpStart プログラムは <code>rootdisk</code> をルートデバイスに設定します。
2	プロファイル内で、 <code>rootdisk</code> が設定されていなくて、 <code>boot_device</code> キーワードが指定されている場合、JumpStart プログラムは <code>rootdisk</code> をブートデバイスに設定します。
3	プロファイル内で <code>rootdisk</code> が設定されていなくて、 <code>filesys cwtxdysz size /</code> エントリが指定されている場合、JumpStart プログラムは <code>rootdisk</code> をエントリで指定されたディスクに設定します。
4	プロファイル内で、 <code>rootdisk</code> が設定されていなくて、 <code>rootdisk.sn</code> エントリが指定されている場合、JumpStart プログラムはシステムのディスクで、(カーネルのプローブ順で) 指定したスライス上の既存のルートファイルシステムを検索します。ディスクが見つかった場合、JumpStart プログラムは見つかったディスクに <code>rootdisk</code> を設定します。
5	プロファイル内で、 <code>rootdisk</code> が設定されていなくて、 <code>partitioning existing</code> が指定されている場合、JumpStart プログラムはシステムのディスクで、(カーネルのプローブ順で) 既存のルートファイルシステムを検索します。ルートファイルシステムが見つからなかった場合、あるいは複数のルートファイルシステムが見つかった場合は、エラーが発生します。ルートファイルシステムが見つかった場合、JumpStart プログラムは見つかったディスクに <code>rootdisk</code> を設定します。
6	プロファイル内で <code>rootdisk</code> が設定されていない場合、JumpStart プログラムは、ルートファイルシステムがインストールされるディスクに <code>rootdisk</code> を設定します。

system_type プロファイルキーワード

`system_type type_switch`

`system_type` は、インストールするシステムのタイプを定義します。

`type_switch` は、オプション `standalone` または `server` を表します。このオプションは、Solaris ソフトウェアをインストールするシステムのタイプを指定するために使用します。`system_type` をプロファイルに指定しないと、デフォルトによって `standalone` が使用されます。

usedisk プロファイルキーワード

`usedisk disk_name ...`

`partitioning default` を指定すると、デフォルトではシステム上のすべての使用可能ディスクが使用されます。`usedisk` プロファイルキーワードには、JumpStart プログラムに使用させる 1 つ以上のディスクを指定します。`disk_name` は、`cxydz` または `cydz` 形式 (たとえば `c0t0d0` または `c0d0s0`) で指定する必要があります。

プロファイルで `usedisk` を指定すると、JumpStart プログラムは `usedisk` キーワードで指定したディスクだけを使用します。

注 – 同じプロファイルに `usedisk` キーワードと `dontuse` キーワードを同時に指定することはできません。

カスタム JumpStart の環境変数

`begin` および `finish` スクリプトには、環境変数を使用できます。たとえば、`begin` スクリプトは、ディスクサイズ (`SI_DISKSIZE`) を抽出し、実際のディスクサイズに基づいてシステムに特定のパッケージをインストールするかどうかを決定できます。

システムについて収集された情報は、これらの環境変数に格納されます。これらの変数は、通常、`rules` ファイルに使用するルールキーワードと値によって設定するかどうかが決まります。

たとえば、システムにすでにインストールされているオペレーティングシステムについての情報は、`installed` キーワードの使用後 `SI_INSTALLED` によってしか入手できません。

表 28-7 は、これらの変数とその値を説明しています。

表 28-7 インストール環境変数

環境変数	値
<code>CHECK_INPUT</code>	JumpStart ディレクトリ内の <code>rules</code> ファイルへのパス。ファイルは <code>/tmp/install_config/rules</code> にマウントされます。
<code>HOME</code>	インストール時のスーパーユーザーのホームディレクトリ (<code>/tmp/root</code>)
<code>PATH</code>	インストール時のシェル検索パス (<code>/sbin:/usr/sbin/install.d:/usr/sbin:/usr/bin</code>)
<code>SI_ARCH</code>	インストールクライアントのハードウェアアーキテクチャー。この変数は、 <code>rules</code> ファイルで <code>arch</code> キーワードを使用する場合に設定します。
<code>SI_BEGIN</code>	<code>begin</code> スクリプトを使用している場合は、その名前
<code>SI_CLASS</code>	インストールクライアントをインストールするために使用されるプロファイルの名前
<code>SI_CONFIG_DIR</code>	JumpStart ディレクトリへのパス。ディレクトリは <code>/tmp/install_config</code> にマウントされます。

表 28-7 インストール環境変数 (続き)

環境変数	値
SI_CONFIG_FILE	JumpStart ディレクトリ内の rules ファイルへのパス。ファイルは /tmp/install_config/rules にマウントされます。
SI_CONFIG_PROG	rules ファイル
SI_CUSTOM_PROBES_FILE	custom_probes.ok ファイル。ここでは、独自のルールキーワードやプローブキーワードを定義できます。custom_probes.ok ファイルを作成して使用すると、デフォルトのルールキーワードのリストを拡張できます (197 ページの「ルールキーワードと値の説明」を参照)。また、このファイルを使用すると、デフォルトのプローブキーワードのリストも拡張できます (228 ページの「プローブキーワードと値」を参照)。
SI_DISKLIST	コマンドで区切られた、インストールクライアント上のディスク名のリスト。この変数は、rules ファイルで disksize キーワードを使用して照合する場合に設定します。SI_DISKLIST および SI_NUMDISKS 変数は、rootdisk (224 ページの「システムのルートディスクを決定する方法」を参照) に使用する物理ディスクを判断するために使用します。
SI_DISKSIZE	コマンドで区切られた、インストールクライアント上のディスクサイズのリスト。この変数は、rules ファイルで disksize キーワードを使用して照合する場合に設定します。
SI_DOMAINNAME	ドメイン名。この変数は、rules ファイルで domainname キーワードを使用して照合する場合に設定します。
SI_FINISH	finish スクリプトを使用する場合はその名前
SI_HOSTADDRESS	インストールクライアントの IP アドレス
SI_HOSTID	インストールクライアントの Ethernet アドレス
SI_HOSTNAME	インストールクライアントのホスト名。この変数は、rules ファイルで hostname キーワードを使用して照合する場合に設定します。
SI_INSTALLED	特定のオペレーティングシステムが入っているディスクのデバイス名 (Solaris、SunOS、System V など)。この変数は、rules ファイルで installed キーワードを使用して照合する場合に設定します。SI_INST_OS と SI_INST_VER は、SI_INSTALLED の値を決定するために使用します。
SI_INST_OS	オペレーティングシステムの名前。SI_INST_OS と SI_INST_VER は、SI_INSTALLED の値を決定するために使用します。
SI_INST_VER	オペレーティングシステムのバージョン。SI_INST_OS と SI_INST_VER は、SI_INSTALLED の値を決定するために使用します。
SI_KARCH	インストールクライアントのカーネルアーキテクチャー。この変数は、rules ファイルで karch キーワードを使用して照合する場合に設定します。
SI_MEMSIZE	インストールクライアントの物理メモリーの量。この変数は、rules ファイルで memsize キーワードを使用して照合する場合に設定します。

表 28-7 インストール環境変数 (続き)

環境変数	値
SI_MODEL	インストールクライアントのモデル名。この変数は、rules ファイルで SI_MODEL キーワードを使用して照合する場合に設定します。
SI_NETWORK	インストールクライアントのネットワーク番号。この変数は、rules ファイルで network キーワードを使用して照合する場合に設定します。
SI_NUMDISKS	インストールクライアントのディスク数。この変数は、rules ファイルで disksize キーワードを使用して照合する場合に設定します。SI_NUMDISKS および SI_DISKLIST 変数は、rootdisk に使用する物理ディスクを決定するために使用します (rootdisk については 224 ページの「システムのルートディスクを決定する方法」を参照)。
SI_OSNAME	Solaris 9 ソフトウェアイメージのオペレーティングシステムリリース。この変数は、たとえば、Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD イメージ上のオペレーティングシステムのバージョンに基づいてシステムに Solaris ソフトウェアをインストールする場合にスクリプト内に指定できます。
SI_PROFILE	マウントされた JumpStart ディレクトリ内のプロファイルへのパス。このパスは /tmp/install_config/profile_name です。派生プロファイルを作成する場合は、SI_PROFILE に /tmp/install.input ファイルが設定されます。
SI_ROOTDISK	論理名 rootdisk によって表されるディスクのデバイス名。この変数は、rules ファイルで disksize または installed キーワードを rootdisk に設定した場合に設定します。
SI_ROOTDISKSIZE	論理名 rootdisk によって表されるディスクのサイズ。この変数は、rules ファイルで disksize または installed キーワードを rootdisk に設定した場合に設定します。
SI_SYS_STATE	/a/etc/.sysidtool.state ファイル。finish スクリプトにこのファイルを指定すれば、システムのレポート時に sysidroot プログラムからスーパーユーザーのパスワードを入力するように求めるプロンプトを表示しないようにすることができます。
SI_TOTALDISK	インストールクライアント上のディスク容量の合計。この変数は、rules ファイルで totaldisk キーワードを使用して照合する場合に設定します。
SHELL	インストール時のデフォルトシェル (/sbin/sh)
TERM	インストールクライアントの端末タイプ
TZ	NIS または NIS+ ネームサービスに指定されているデフォルトの時間帯

プローブキーワードと値

表 28-8 は、各ルールキーワードと対応するプローブキーワードを説明しています。

注 - プローブキーワードは、rules ファイルの最初、またはその近くに指定してください。

表 28-8 プローブキーワードの説明

ルールキーワード	対応するプローブキーワード	プローブキーワードの説明
any	なし	
arch	arch	カーネルアーキテクチャー (SPARC) を判断して SI_ARCH を設定します。
disksize	disks	システムのディスクサイズ (M バイト) をカーネルプローブ順 (c0t3d0s0, c0t3d0s1, c0t4d0s0) で返します。disksize は、SI_DISKLIST、SI_DISKIZES、SI_NUMDISKS、および SI_TOTALDISK を設定します。
domainname	domainname	システムの NIS または NIS+ ドメイン名、あるいは空白を返して、SI_DOMAINNAME を設定します。domainname キーワードは domainname (1M) の出力を返します。
hostaddress	hostaddress	システムの IP アドレス (lo0 ではない ifconfig (1M) -a の出力にリストされた最初のアドレス) を返して、SI_HOSTADDRESS を設定します。
hostname	hostname	システムのホスト名 (uname (1) -n からの出力) を返して、SI_HOSTNAME を設定します。
installed	installed	システムにインストールされた Solaris オペレーティング環境のバージョン名を返して、SI_ROOTDISK と SI_INSTALLED を設定します。 Solaris リリースは検出されたが、バージョンを判断できないという場合は、返されるバージョンは SystemV になります。
karch	karch	システムのプラットフォームグループ (sun4m、および sun4 など) を返し、SI_KARCH を設定します。プラットフォーム一覧については、『Solaris 9 Sun ハードウェアマニュアル』を参照してください。
memsize	memsize	システム上の物理メモリーのサイズ (M バイト) を返して、SI_MEMSIZE を設定します。
model	model	システムのプラットフォーム名を返して、SI_MODEL を設定します。プラットフォーム名のリストについては、『Solaris 9 Sun ハードウェアマニュアル』を参照してください。
network	network	システムのネットワーク番号を返します。これは JumpStart プログラムがシステムの IP アドレスとサブネットマスクの論理和をとって判断します。システムの IP アドレスとサブネットマスクは、lo0 ではない ifconfig (1M) -a 出力にリストされた最初のアドレスから抽出されます。network キーワードは SI_NETWORK も設定します。

表 28-8 プローブキーワードの説明 (続き)

ルールキーワード	対応するプローブ キーワード	プローブキーワードの説明
osname	osname	CD で検出された Solaris オペレーティング環境のバージョンおよびオペレーティングシステム名を返して、SI_OSNAME を設定します。 Solaris リリースは検出されたが、バージョンを判断できないという場合は、返されるバージョンは SystemV になります。
	rootdisk	システムのルートディスクの名前とサイズ (M バイト) を返して、SI_ROOTDISK を設定します。
totaldisk	totaldisk	システム上のディスク領域の合計 (M バイト) を返して、SI_TOTALDISK を返します。ディスク容量の合計には、システムに接続された操作可能なディスクすべてが含まれます。

第 29 章

Solaris Live Upgrade (トピック)

この章では、Solaris Live Upgrade を使って非アクティブブート環境を作成し、これをアップグレードしてアクティブブート環境に切り替える方法について説明します。

第 30 章	Solaris Live Upgrade の概要情報を提供します。
第 31 章	ブート環境の作成に必要な情報を提供します。
第 32 章	Solaris Live Upgrade のインストール方法、メニューの使用方法、ブート環境の作成方法について説明します。
第 33 章	オペレーティングシステムのアップグレード方法、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、ブート環境をアクティブにする方法、アップグレード時に障害が発生した場合のスムーズな修復方法について説明します。
第 34 章	ブート環境の管理と状態使用の表示方法について説明します。
第 35 章	Solaris Live Upgrade の実例を提示します。
第 36 章	Solaris Live Upgrade コマンドの一覧を提示します。

第 30 章

Solaris Live Upgrade の概要

この章では、Solaris Live Upgrade の実行手順について説明します。

注 - このマニュアルでは「スライス」という用語を使用しますが、一部の Solaris のマニュアルとプログラムでは、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。

Solaris Live Upgrade の紹介

Solaris Live Upgrade を利用してオペレーティングシステムをアップグレードすると、従来アップグレードのために費やしていたサービス停止時間を大幅に短縮できます。このソフトウェアでは、現在動作しているブート環境を複製し、その複製元のブート環境を動作させたまま複製された環境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可能です。環境をアップグレードしても、フラッシュ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。これらの処理の後、システムをリポートすると、複製のブート環境がアクティブ化されて、アクティブなブート環境になります。何か問題が発生しても、対応できるようになっています。リポートするだけで元のブート環境に戻すことができるので、テストや評価処理のためのサービス停止時間を削減することができます。

Solaris 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、ブート環境のコピーを作成して、次のような作業を行うことができます。

- システムのアップグレード
- 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）の変更

- 異なるイメージを持つ複数のブート環境の保守。たとえば、現在のパッチを持つブート環境を作成して、同時に、Update リリースを持つブート環境を作成できます。

Solaris Live Upgrade を使用するには、システム管理についての基礎的な事柄を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、システム管理作業に関する基本的な情報については、『Solaris のシステム管理 (基本編)』を参照してください。

Solaris Live Upgrade の実行手順

次に、非アクティブなブート環境を作成して、この環境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ために必要な作業の概要を示します。

ブート環境の作成 (概要)

ブート環境を作成すると、クリティカルな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、クリティカルな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概要

Solaris Live Upgrade はファイルシステムをクリティカルなファイルシステムと共有可能なファイルシステムの 2 種類に区別します。クリティカルなファイルシステムとは、Solaris オペレーティング環境に必須であり、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 `vfstab` において別々のマウントポイントを持つファイルシステムのことです。たとえば、ルート (`/`)、`/usr`、`/var`、または `/opt` がクリティカルな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常に、ソースブート環境から非アクティブなブート環境にコピーされます。クリティカルなファイルシステムのことを「共有不可能」と呼ぶこともあります。共有可能なファイルシステムとは、`/export` 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 `vfstab` 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。共有可能なファイルシステムについての詳細は、253 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。

スワップは、共有可能なファイルシステムの中でも特別なものです。他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。このためには、キャラクタ型ユーザーインタフェースま

たはコマンド行を使用して、`lucreate(1M)` に `-m` オプションを指定して実行します。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、スワップスライスは使用中であってはならないという制限があります。スワップスライスが他のブート環境によって使用されている場合、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、ブート環境の作成は失敗します。スワップスライスは必須ではありません。スワップを構成し直す手順については、「ブート環境を作成する (キャラクタインタフェース)」の手順 9、または 271 ページの「ブート環境を作成しスワップを再構成する (コマンド行インタフェース)」を参照してください。

注 – ブート環境のファイルシステムを作成する場合のルールは、Solaris オペレーティング環境のファイルシステムを作成する場合と同じです。Solaris Live Upgrade では、クリティカルな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` コマンドを用いて、ルート (`/`) と `/kernel` を別々のファイルシステムに作成することができますが、このようにルート (`/`) を分割するのは誤りです。

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、クリティカルなファイルシステムを他のスライスにコピーする必要があります。新しいブート環境を作成する前には、ディスクを準備する必要がある場合もあります。ディスクをチェックして、適切にフォーマットされていることを確認します。

- 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、共有したい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したい場合、新しいブート環境を作成して、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。この時点で、この新しいスライスは将来のブート環境と共有できます。異なる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、253 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。

新しいブート環境を作成するには、まず、クリティカルな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。メニューからスライスをフォーマットする手順については、「ブート環境を作成する (キャラクタインタフェース)」の手順 6 を参照してください。

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。

ファイルシステムを分割およびマージする手順については、次の節を参照してください。

- 「ブート環境を作成する (キャラクタインタフェース)」の手順 8 または手順 7
- 269 ページの「ブート環境を作成しファイルシステムを分割する (コマンド行インタフェース)」
- 267 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 (コマンド行インタフェース)」

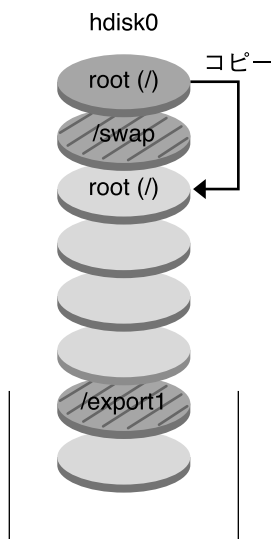
非アクティブなブート環境でファイルシステムを構成した後は、自動コピーを開始します。クリティカルなファイルシステムは、指定された宛先ディレクトリにコピーされます。共有可能なファイルシステムは (それらの一部をコピーするように指定しない限り) コピーされずに共有されます。ファイルシステムをアーカイブから非アクティブなブート環境にコピーするとき、ファイルは新たに定義されたディレクトリにコピーされるので、アクティブなブート環境は変更されません。新しいブート環境の作成手順については、259 ページの「新しいブート環境の作成」を参照してください。

次の図に、さまざまな方法で新しいブート環境を作成する例を示します。

- 図 30-1 に、クリティカルなファイルシステムのルート (/) を同じまたは別のディスク上のスライスにコピーして、新しいブート環境を作成する方法を示します。アクティブなブート環境は、既存のスライス上にルート (/) 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 (/) の複製を持っています。ファイルシステム /swap および /export/home 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。

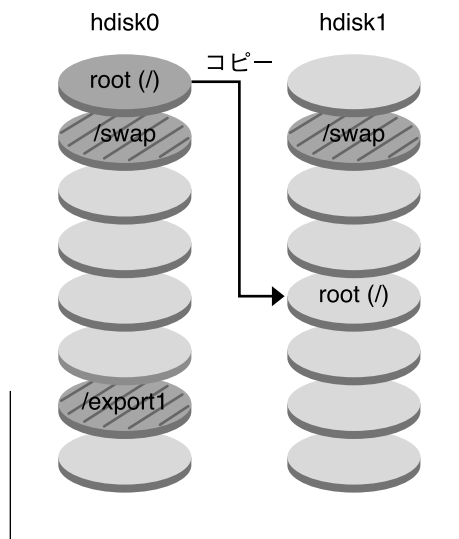
ブート環境の作成 - ルート(/) からルート(/) へのコピー

ハードディスク1台



アップグレード

ハードディスク2台



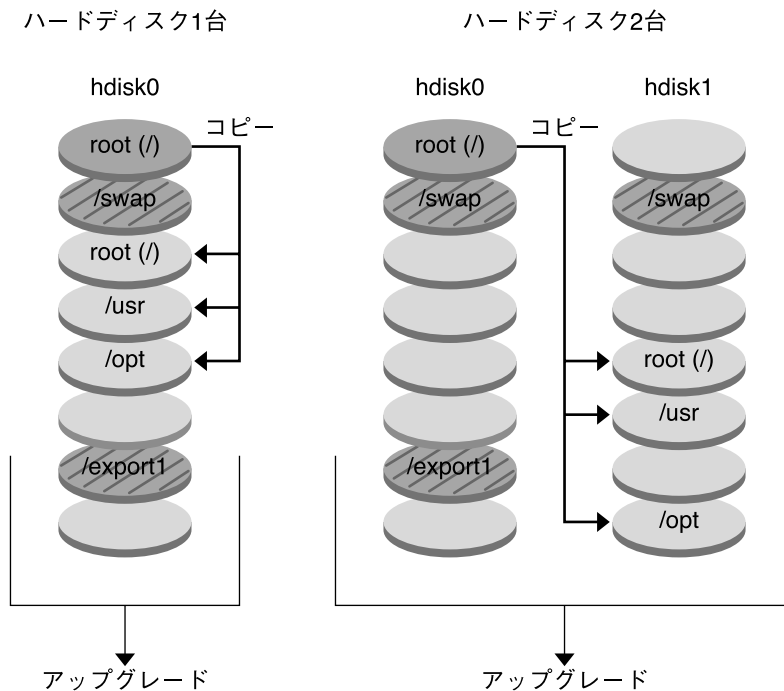
アップグレード

- 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

図 30-1 非アクティブなブート環境の作成 — ルート (/) からルート (/) へのコピー

- 図 30-2 に、クリティカルなファイルシステムを同じまたは別のディスク上の複数のスライスに分割およびコピーして、新しいブート環境を作成する方法を示します。アクティブなブート環境は、既存のスライス上にルート (/) を持っています。このスライスでは、ルート (/) 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 (/) は分割され、/usr と /opt は別のスライスに配置されています。ファイルシステム /swap と /export/home は両方のブート環境で共有されます。

ブート環境の作成 - ファイルシステムの分割

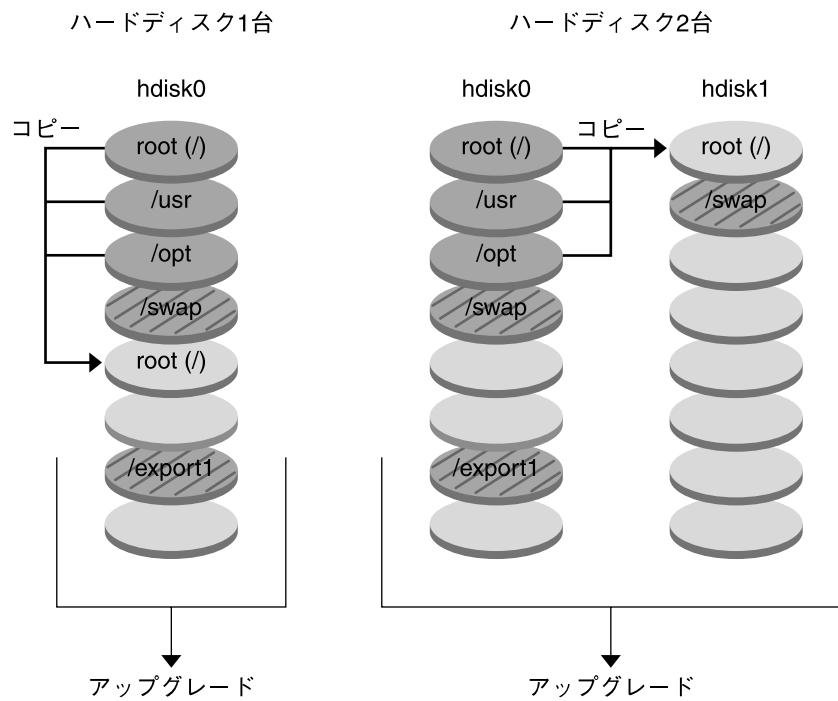


- 現在のリリース X
クリティカルなファイルシステムのルート(/)
- 非アクティブなリリース X
クリティカルなファイルシステムのルート (/) /usr /opt
- 共有ファイルシステム

図 30-2 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

- 図 30-3 に、クリティカルなファイルシステムをマージし、同じまたは別のディスク上のスライスにコピーして、新しいブート環境を作成する方法を示します。アクティブなブート環境では、ルート (/)、/usr、/var、および /opt がそれぞれ別のスライス上にあります。新しいブート環境では、/usr と /opt はルート (/) と同一のスライス上にマージされます。ファイルシステム /swap と /export/home は両方のブート環境で共有されます。

ブート環境の作成 - ファイルシステムのマージ



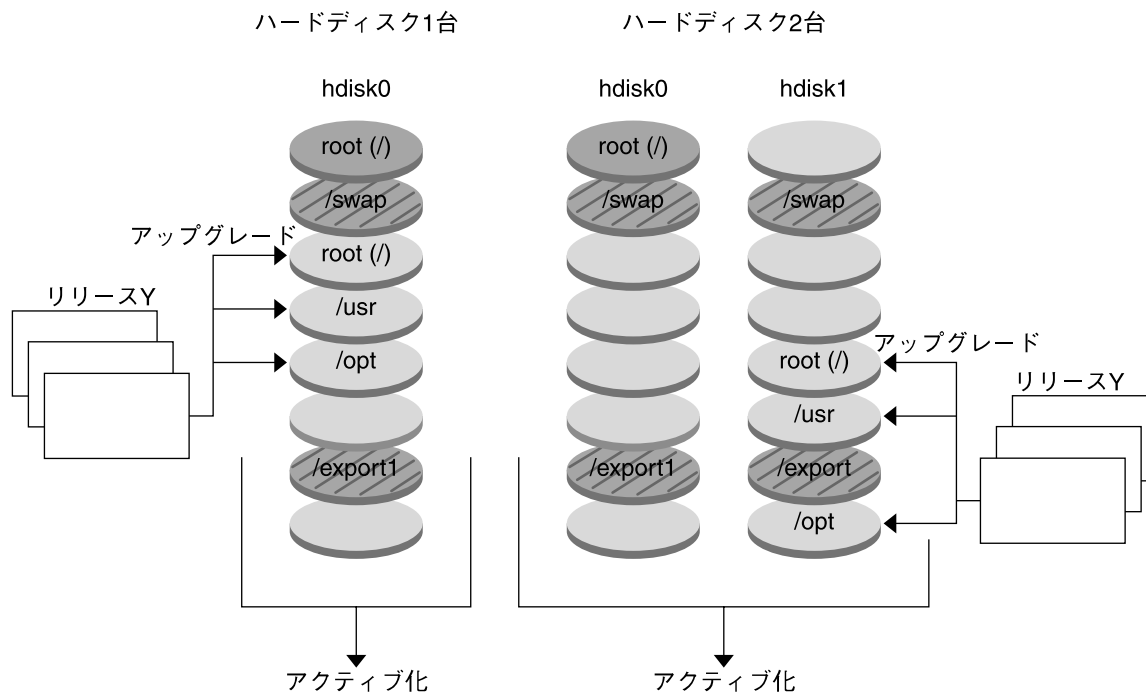
- 現在リソース X
クリティカルなファイルシステムのルート (/) /usr /opt
- 非アクティブなリリース Y
クリティカルなファイルシステムのルート (/)
- ▣ 共有ファイルシステム

図 30-3 非アクティブなブート環境の作成 — ファイルシステムのマージ

ブート環境のアップグレード (概要)

ブート環境を作成した後、アップグレードを行うまでそのブート環境は変更されません。ブート環境のアップグレードはいつでも実行できます。アップグレードを行っても、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。準備が整った時点で、新しいリリースをアクティブにします。図 30-4 に、非アクティブなブート環境のアップグレードの例を示します。ブート環境のアップグレード手順については、第 33 章を参照してください。

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

図 30-4 非アクティブなブート環境のアップグレード

アップグレードのかわりにフラッシュアーカイブを用いてブート環境にインストールすることもできます。フラッシュインストール機能を利用すると、マスターシステムと呼ばれるシステム上に Solaris オペレーティング環境のリファレンスインストールを 1 つ作成できます。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。フラッシュインストール機能についての詳細は、第 16 章を参照してください。

フラッシュアーカイブをシステムにインストールするとき、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。図 30-5 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。フラッシュアーカイブのインストール手順については、286 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。

フラッシュアーカイブのインストール

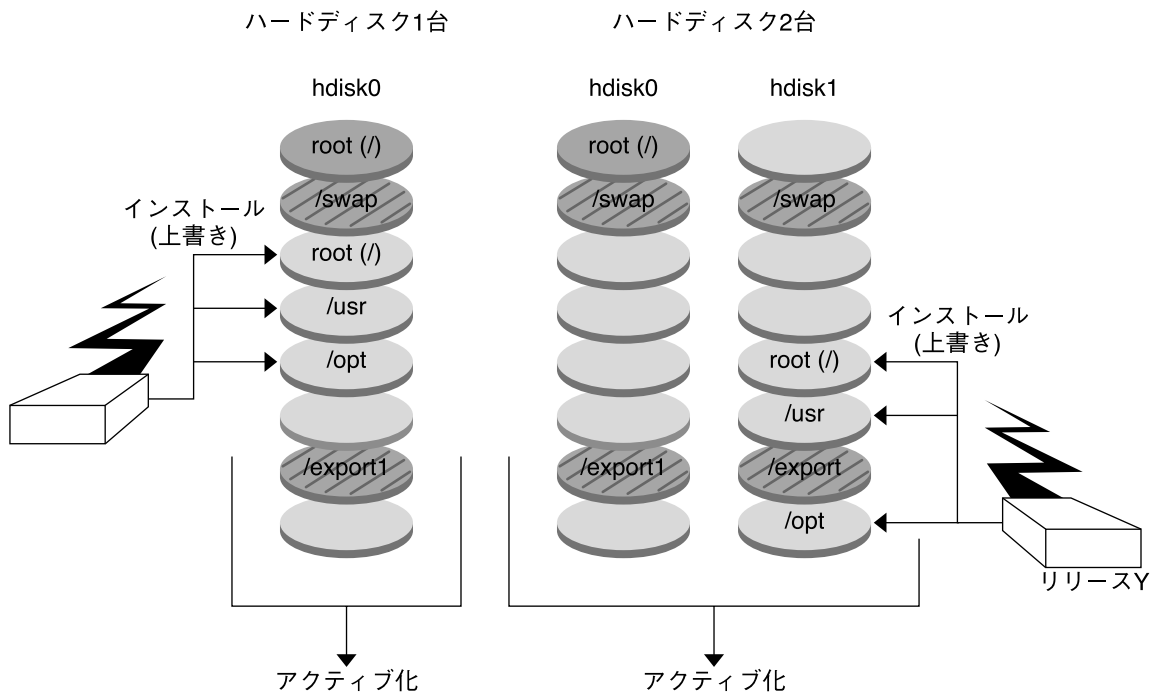


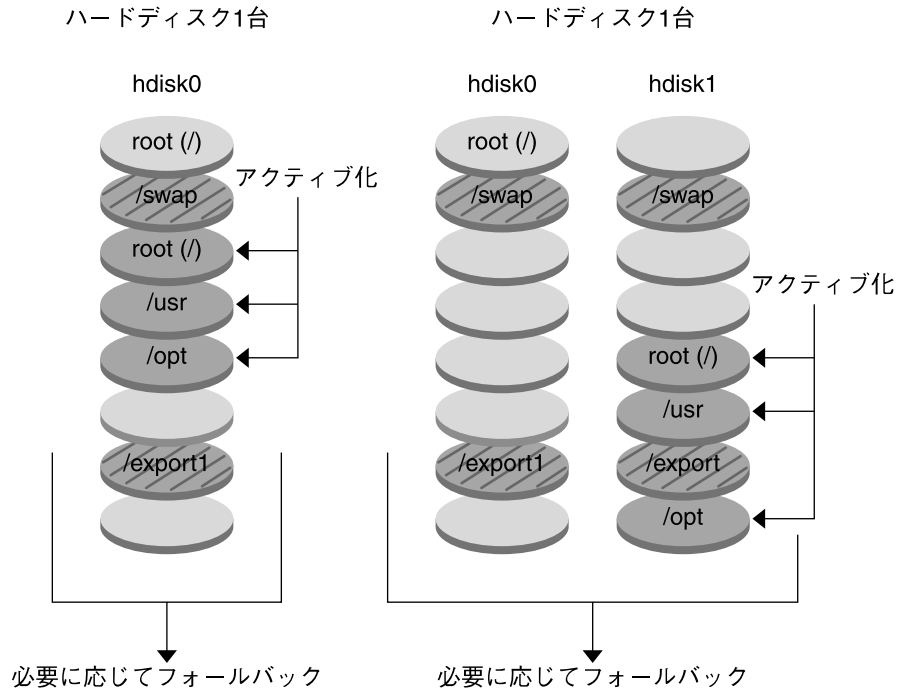
図 30-5 フラッシュアーカイブのインストール

ブート環境のアクティビ化 (概要)

新しいブート環境に切り換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アクティブにしてリポートするだけです。新しいブート環境をアクティブにすると、そのブート環境はブート可能になります。新たに作成したブート環境を初めて起動するとき、ブート環境間でファイルの同期がとられます。ここで「同期」とは、最

後にアクティブであったブート環境からこれからブートするブート環境に、システムファイルとシステムディレクトリを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なブート環境となります。図 30-6 に、リブート後に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に切り替わる様子を示します。ブート環境をアクティブにする手順については、291 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

図 30-6 非アクティブなブート環境のアクティブ化

元のブート環境へのフォールバック (概要)

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。フォールバックを実行する必要があるのは、新しいブート環境がブートできない場合、ブートできるが正常に稼動しない場合、あるいは、ブート結果に満足できない場合です。

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。フォールバックを実行できるのは、`luactivate` を使用して新しいブート環境をアクティブにしたブート環境だけです。

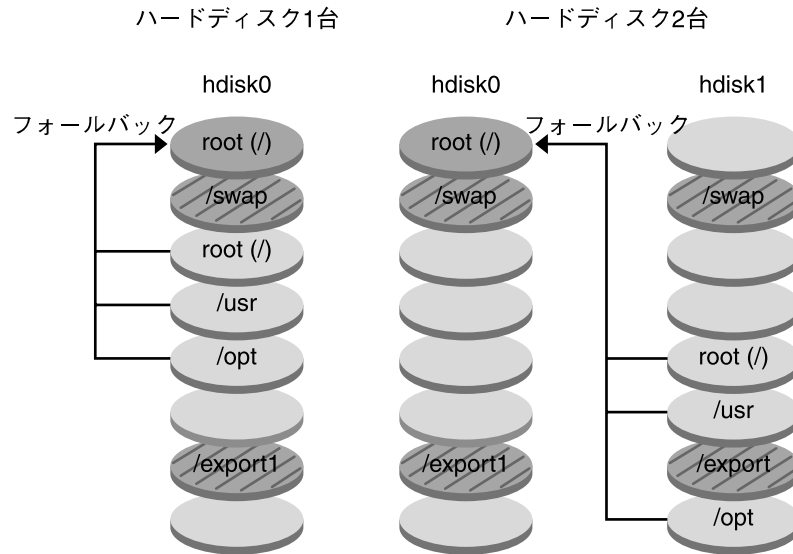
以前のブート環境にフォールバックするには、次の手順に従います。

- 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場合、`luactivate` 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。
- 新しいブート環境がブートしない場合、フォールバックしたいブート環境をシングルユーザーモードでブートし、`luactivate` 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場合、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て、フォールバックしたいブート環境上のルート (/) ファイルシステムをマウントし、`luactivate` コマンドを実行して、リブートします。

フォールバックの手順については、294 ページの「問題の解決: 元のブート環境へのフォールバック (コマンド行インタフェース)」を参照してください。

図 30-7 に、リブートしてフォールバックしたときにブート環境が切り替わる様子を示します。

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 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

図 30-7 元のブート環境へのフォールバック

ブート環境の保守 (概要)

ブート環境の名前変更や削除など、さまざまな保守作業も行うことができます。保守作業の手順については、第 34 章を参照してください。

第 31 章

Solaris Live Upgrade の計画

この章では、Solaris Live Upgrade のインストールと使用を開始する前に考慮すべきガイドラインと要件を説明します。42 ページの「アップグレード用のチェックリスト」でアップグレード全般に関する情報も確認してください。この章の内容は次のとおりです。

- 247 ページの「Solaris Live Upgrade のシステム要件」
- 248 ページの「Solaris Live Upgrade のディスク容量の要件」
- 248 ページの「Solaris Live Upgrade によるパッケージおよびパッチの管理」
- 251 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」
- 254 ページの「リモートシステムからの Solaris Live Upgrade の使用」

Solaris Live Upgrade のシステム要件

- Solaris 2.6、Solaris 7、または Solaris 8 オペレーティング環境から Solaris 8 または Solaris 9 オペレーティング環境へアップグレードできます。
- Solaris 7 オペレーティング環境にはアップグレードできません。
- 現在実行中のシステムにインストールした Solaris Live Upgrade が収められているリリースにアップグレードできます。たとえば、現在のオペレーティング環境のみに Solaris 9 リリースの Solaris Live Upgrade をインストールした場合、Solaris 9 リリースにのみアップグレード可能です。

Solaris Live Upgrade は Solaris 9 ソフトウェアに含まれていますが、Solaris 9 以前のリリースからアップグレードしたい場合は、現在のオペレーティング環境に Solaris Live Upgrade のパッケージをインストールする必要があります。Solaris Live Upgrade パッケージのインストールには、次のものを使用します。

- Solaris 9 DVD、Solaris 9 SOFTWARE 2 of 2 CD、またはネットインストールイメージ上にあるインストーラ。

- pkgadd コマンド。pkgadd コマンドを使用する場合、パッケージは SUNWlur、SUNWluu の順番でインストールしてください。

Solaris Live Upgrade ソフトウェアのインストール方法については、257 ページの「Solaris Live Upgrade をインストールする方法」を参照してください。

Solaris Live Upgrade のディスク容量の要件

アップグレードの一般的なディスク容量の要件に従います。第 5 章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てください。サイズが計算されたところで、処理を中断できません。

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確認するには、各システムのマニュアルを参照してください。

Solaris Live Upgrade によるパッケージおよびパッチの管理

以下の各節では、Solaris Live Upgrade に必要なパッケージを示し、推奨パッチ情報を提供します。Solaris Live Upgrade を使ってパッケージおよびパッチを追加する方法については、282 ページの「Solaris Live Upgrade によるパッケージやパッチの管理」を参照してください。



注意 – パッケージやパッチのアップグレード、追加、削除を行うためには、そのパッケージまたはパッチが SVR4 Advanced Packaging Guideline に準拠していなければなりません。サンのパッケージはこのガイドラインに準拠していますが、サン以外のベンダーのパッケージについては保証していません。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。

Solaris Live Upgrade によるパッケージの追加と削除の詳細については、luupgrade (1M) のマニュアルページを参照してください。パッケージの要件の詳細については、付録 C を参照してください。

必要なパッケージ

次の表は、Solaris Live Upgrade を使用する上で必要なパッケージを示しています。この表で、現在のオペレーティング環境に必要なパッケージを確認してください。使用しているリリースに必要なパッケージ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それらを追加してください。

表 31-1 Solaris Live Upgrade に必要なパッケージ

Solaris 2.6 リリース	Solaris 7 リリース	Solaris 8 リリース
SUNWadmap	SUNWadmap	SUNWadmap
SUNWadmfw	SUNWadm	SUNWadm
SUNWadm	SUNWlibC	SUNWlibC
SUNWmfrun		SUNWbzip
SUNWloc		
SUNWlibC		

▼ システム上のパッケージを確認するには

- システム上に存在するパッケージを表示するには、次のように入力します。

```
% pkginfo [package_name]
```

```
package_name          確認したいパッケージ名
```

システムパッチレベルのチェック

Solaris Live Upgrade ソフトウェアは、複数の Solaris オペレーティング環境バージョンでインストールと実行ができるように設計されています。Solaris Live Upgrade の処理が正しく行われるようにするためには、各 OS バージョン用に提供されている最新の推奨パッチとセキュリティパッチを適用する必要があります。実行中の Solaris のリリースに適したパッチクラスタのリビジョンについては、<http://sunsolve.sun.com> を参照してください。

lucreate コマンドを用い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クリティカルファイルシステムがある場合、このファイルシステムは、`-m` オプションで指定しなくても新しく作成されたファイルシステムの上位 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したい場合は、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、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の規則は、Solaris オペレーティング環境のファイルシステムを作成する場合と同じです。Solaris 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できてしまいます。たとえば、lucreate コマンドを用いて、ルート (/) と /kernel を別々のファイルシステムに作成することも可能です。しかし、これは、無効な分割方式です。

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。

Solaris Live Upgrade を正しく機能させるには、アクティブブート環境の vfstab ファイルの内容が有効で、少なくとも ルート (/) のエントリが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確認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。スライスは、次の条件を満たしていなければなりません。

- 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
- Veritas VxVM ボリュームではない
- 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
- sun4m システムの場合、ルート (/) ファイルシステムは最大 2G バイトまでである。

「Choices」メニューには、非アクティブブート環境の作成に使用できる空きスライスが表示されます。しかし、スライスの中には、Veritas VxVM ボリュームや Solaris ボリュームマネージャメタデバイスのように、未使用であるが「Choices」メニューに表示されないものがあります。

ルート (/) のミラーとメタデバイスを選択するための指針

Solaris Live Upgrade は、Solaris ボリュームマネージャメタデバイスまたは Veritas ボリュームマネージャ VxVM ボリュームを使用しているシステム上で使用できます。ソースブート環境に含まれる物理ディスクスライス、Solaris ボリュームマネージャメタデバイス、Veritas ボリュームマネージャボリュームは自由に組み合わせることができます。新しいブート環境を作成する際は、物理ディスクスライスまたは Solaris ボリュームマネージャメタデバイスのどちらかをこの環境のルート (/) ファイルシステムとして選択できます。Solaris ボリュームマネージャメタデバイスを選択した場合は、単一のディスクまたは単一のディスクストライプ上のミラーを使って、このメタデバイスをストライプ化する必要があります。詳細については、`metaroot (1M)` のマニュアルページを参照してください。

Veritas VxFS ボリュームは、新しいブート環境のルート (/) ファイルシステムとして選択できません。ルート (/) ファイルシステム以外のファイルシステムとしてであれば、物理ディスクスライス、Solaris ボリュームマネージャメタデバイス、Veritas VxFS ボリュームのいずれかを使用できます。

表 31-2 は、ブート環境を作成するための有効なディスク構成について説明しています。メタデバイスを使用する場合とボリュームを使用する場合とでは、有効なディスク構成が異なります。

表 31-2 メタデバイスまたはボリュームの有効なスライス

製品	ソーススライス	ターゲットがルート (/) ファイルシステムのメタデバイスである場合	ターゲットがルート (/) ファイルシステムではなく、 <code>/usr</code> 、 <code>/var</code> 、 <code>/opt</code> のいずれかである場合
Solaris ボリュームマネージャ	ルート (/) ファイルシステムの場合、ソースはメタデバイスか物理スライス	ルート (/) ファイルシステムは単一ディスクまたは単一ディスクストライプ上のミラーを持つストライプでなければならない	その他のファイルシステムは物理スライスでもメタデバイスでもよい
Veritas VxVM ボリュームマネージャ	ルート (/) ファイルシステムの場合、ソースはボリュームか物理スライス	ルート (/) ファイルシステムは VsVM ボリュームにはできない。ルート (/) は必ず物理スライス	その他のファイルシステムは物理スライスでもボリュームでもよい

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

- 物理スライス (`/dev/dsk/cnum tnumdnums num`)
- Solaris ボリュームマネージャメタデバイス (`/dev/md/dsk/d num`)
- Veritas VxFS ボリューム (`/dev/vx/dsk/ volume_name`)

ブート環境でメタデバイスまたはボリュームを使用する場合、フラッシュアーカイブのアップグレードまたはインストール時に特別に注意すべき項目があります。詳細については、281 ページの「メタデバイスやボリュームのアップグレード」を参照してください。

注 – Veritas VxVM のアップグレードで問題が生じる場合は、353 ページの「Veritas VxVm の実行中に Solaris 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」を参照してください。

swap ファイルシステムのスライスを選択するための指針

スワップスライスは、現在のブート環境 (-s オプションを使用する場合はソースブート環境) 以外のブート環境で使用することはできません。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、スワップスライスがほかのブート環境で使用されている場合、ブート環境の作成は失敗します。

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

Solaris Live Upgrade は、スライスの内容を指定の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したい場合は、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(/)、/var のようなオペレーティング環境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。

- 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(/)、/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。
- ディレクトリを共有したい場合、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もと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 8 から Solaris 9 にアップグレードし、両方のオペレーティング環境で /home を共有したい場合は、`lucreate -m` コマンドを実行して、/home 固有のスライス上に独立したファイルシステムとして配置し、Solaris 8 リリースを作成します。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。さらに、この複製ブート環境を Solaris 9 リリースへアップグレードします。/home は Solaris 8 リリースと Solaris 9 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、234 ページの「ブート環境の作成 (概要)」を参照してください。

リモートシステムからの Solaris Live Upgrade の使用

キャラクタインタフェースを `tip` 回線などを介してリモートで表示する場合は、必要に応じて `TERM` 環境変数を `VT220` に設定してください。また、共通デスクトップ環境 (CDE) を使用する場合は、`TERM` 変数の値を `xterm` ではなく `dtterm` に設定してください。

第 32 章

Solaris Live Upgrade によるブート環境の作成

この章では、Solaris Live Upgrade のインストール、メニューの使用、およびブート環境の作成について説明します。この章の内容は次のとおりです。

- 255 ページの「Solaris Live Upgrade インタフェースの概要」
- 256 ページの「Solaris Live Upgrade メニューの使用」
- 257 ページの「作業マップ: Solaris Live Upgrade のインストールとブート環境の作成」
- 257 ページの「Solaris Live Upgrade のインストール」
- 258 ページの「Solaris Live Upgrade の起動と停止 (キャラクタインタフェース)」
- 259 ページの「新しいブート環境の作成」

Solaris Live Upgrade インタフェースの概要

Solaris Live Upgrade は、キャラクタユーザーインタフェース (CUI) を介して使用することもコマンド行インタフェース (CLI) で使用することも可能です。操作内容と手順は、CUI の場合と CLI の場合とでほぼ同じです。以下の説明では、CUI と CLI の両方の場合について手順を示してあります。CUI では、Solaris Live Upgrade の一部の機能にアクセスできません。コマンド一覧については第 36 章を参照してください。また、コマンドの詳しい説明については各マニュアルページを参照してください。

日本語環境では CUI および CLI のメッセージは、翻訳されていません。

Solaris Live Upgrade メニューの使用

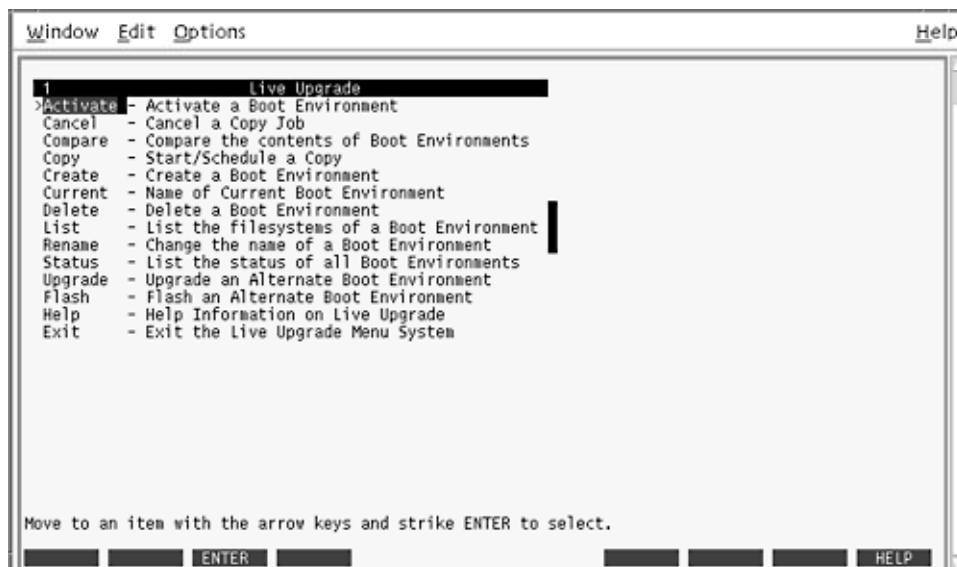


図 32-1 Solaris Live Upgrade のメインメニュー

Solaris Live Upgrade キャラクターユーザーインターフェースのメニュー間の移動には、矢印キーとファンクションキーを使用します。上下に移動する場合やフィールド内にカーソルを置く場合は、矢印キーを使用してください。処理を実行する場合は、ファンクションキーを使用してください。メニューの最下部には、キーボード上のファンクションキーを示す黒い矩形が表示されます。これらは、最初の矩形が F1、2 番目の矩形が F2 という順序で並んでいます。有効な矩形には、「Save」などの処理を示す語句が表示されています。「Configuration」メニューでは、矩形ではなくファンクションキーの番号と対応する動作が表示されます。

- F3 を使用すると、どのメニューの場合も作業の保存 (SAVE) が行われてそのメニューが終了します。
- F6 を使用すると、どのメニューにおいても作業が取り消され (CANCEL)、変更の保存が行われないままメニューが終了します。
- ほかのファンクションキーの動作は、メニューによって異なります。

キーボード上のファンクションキーが Solaris Live Upgrade メニュー上のファンクションキーと適切に対応していない場合は、以下に説明する作業においてファンクションキーを押すように指示されている箇所、Control-F キーおよび該当する番号を使用してください。

作業マップ: Solaris Live Upgrade のインストールとブート環境の作成

表 32-1 作業マップ: Solaris Live Upgrade の使用

作業	説明	参照先
Solaris Live Upgrade パッケージのインストール	Solaris 2.6、Solaris 7、または Solaris 8 オペレーティング環境にパッケージをインストールします。	257 ページの「Solaris Live Upgrade のインストール」
Solaris Live Upgrade の起動	Solaris Live Upgrade のメインメニューを起動します。	258 ページの「Solaris Live Upgrade の起動と停止 (キャラクターインタフェース)」
ブート環境の作成	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。	259 ページの「新しいブート環境の作成」

Solaris Live Upgrade のインストール

Solaris 2.6、Solaris 7、または Solaris 8 オペレーティング環境からアップグレードする場合は、現在のオペレーティング環境に Solaris Live Upgrade パッケージをインストールする必要があります。

▼ Solaris Live Upgrade をインストールする方法

1. Solaris 9 DVD または Solaris 9 SOFTWARE 2 of 2 CD を挿入します。
2. インストーラを実行します。
 - Solaris 9 DVD を使用する場合は、インストーラのあるディレクトリに移動してからインストーラを実行します。

```
# cd /cdrom/cdrom0/Solaris_9/Tool/Installers
# ./liveupgrade20
```

Solaris Web Start インストーラが表示されます。
 - Solaris 9 SOFTWARE 2 of 2 CD を使用する場合は、インストーラを実行します。

```
% ./installer
```

Solaris Web Start インストーラが表示されます。

3. 「インストール形式の選択 (Select Type of Install)」パネルで「カスタム (Custom)」をクリックします。
4. 「ロケールの選択 (Locale Selection)」パネルで、インストールする言語をクリックします。
5. 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 (Component Selection)」パネルの「次へ (Next)」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 (Product Selection)」パネルの Solaris Live Upgrade の項目で「デフォルトインストール (Default Install)」をクリックします。
6. Solaris Web Start の指示に従って、ソフトウェアをインストールします。

Solaris Live Upgrade の起動と停止 (キャラクタインタフェース)

Solaris Live Upgrade のメニュープログラムの起動と停止方法について説明します。

▼ Solaris Live Upgrade メニューを起動する

注 - キャラクタインタフェースを tip 回線などを介してリモートで表示する場合は、必要に応じて TERM 環境変数を VT220 に設定してください。また、共通デスクトップ環境 (CDE) を使用する場合は、TERM 変数の値を xterm ではなく dtterm に設定してください。

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

```
# /usr/sbin/lu
```

Solaris Live Upgrade のメインメニューが表示されます。

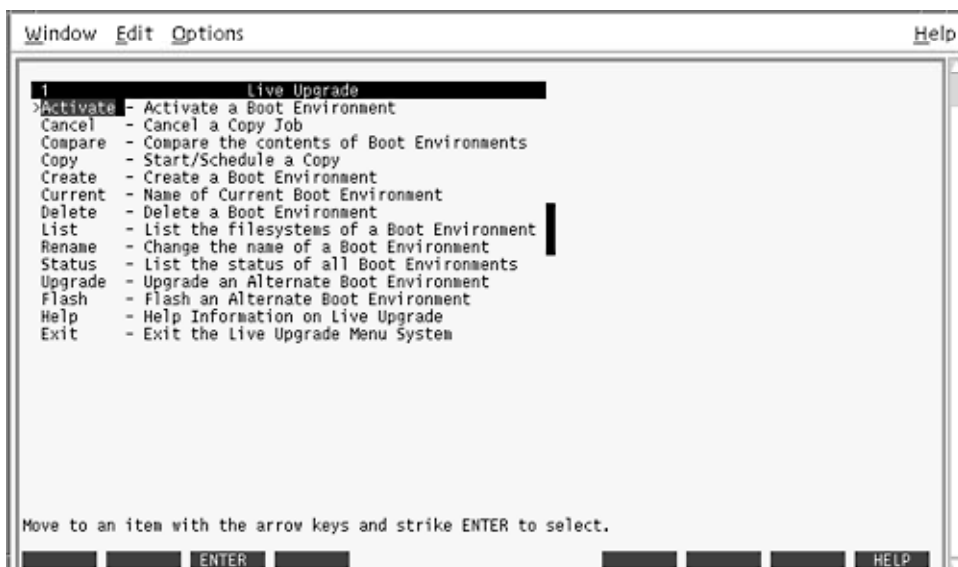


図 32-2 Solaris Live Upgrade のメインメニュー

▼ Solaris Live Upgrade のメニューを終了するには

- メインメニューの「Exit」を選択します。

新しいブート環境の作成

Solaris Live Upgrade は、メニューを介して使用することもコマンド行インタフェース (CLI) で使用することもできます。以下の説明では、両方のインタフェースについて手順を説明しています。これらの手順では、Solaris Live Upgrade の使用に関する詳しい説明は省略しています。コマンドの詳細については、第 36 章を参照してください。また、コマンド行インタフェースの詳細については、各マニュアルページを参照してください。

ブート環境を作成すると、アクティブブート環境から新しいブート環境にクリティカルファイルシステムをコピーできます。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、「Create」メニュー、「Configuration」サブメニュー、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(共有可能なファイルシステム)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、234 ページの「ブート環境の作成(概要)」を参照してください。

この節では次の手順を説明します。

- 260 ページの「ブート環境を作成する(キャラクタインタフェース)」
- 265 ページの「ブート環境を初めて作成する(コマンド行インタフェース)」
- 267 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(コマンド行インタフェース)」
- 269 ページの「ブート環境を作成しファイルシステムを分割する(コマンド行インタフェース)」
- 271 ページの「ブート環境を作成しスワップを再構成する(コマンド行インタフェース)」
- 273 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(コマンド行インタフェース)」
- 275 ページの「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(コマンド行インタフェース)」
- 276 ページの「別々のソースから単一のブート環境を作成(コマンド行インタフェース)」
- 277 ページの「フラッシュアーカイブ用ブート環境の作成(コマンド行インタフェース)」

▼ ブート環境を作成する(キャラクタインタフェース)

1. メインメニューから「**Create**」を選択します。
「Create a Boot Environment」サブメニューが表示されます。
2. アクティブブート環境(必要に応じて)と新しいブート環境の名前を入力し、確定します。アクティブブート環境の名前の入力が必要なのは、最初にブート環境を作成するときだけです。

ブート環境名は英数字で 30 文字以内とします。使用できるのは英数文字だけで、マルチバイト文字を使用することはできません。

```
Name of Current Boot Environment:    solaris8
Name of New Boot Environment:       solaris9
```

3. **F3** を押して変更を保存します。
「Configuration」メニューが表示されます。

Active Boot Environment - solaris8				
Mount Point	Device	FS Type	Size (MB)	% Used
/	c0t0d0s0	ufs	824	74
-	c0t0d0s1	swap	257	0

New Boot Environment - solaris9				
Mount Point	Device	FS Type	Size (MB)	Recommended Min Size (MB)
/		ufs		1025
-	c0t0d0s1	swap	257	3

Esc	F2	F3	F4	F5	F6	F7	F8	F9	^D	^X
HELP	CHOICE	SAVE	SLICE	PRINT	CANCEL	SCHEDULE	SPLIT	MERGE	CLR	OTHR

図 32-3 Solaris Live Upgrade の「Configuration」メニュー

「Configuration」メニューには次の項目があります。

- 元のブート環境は、画面の上部に表示されます。画面の下部には作成されたブート環境が表示されます。
- 「Device」フィールドには次の情報が含まれています。
 - ディスクデバイス名。/dev/dsk/c numdnumtnums num の形式で表示されます。
 - Solaris ボリュームマネージャのメタデバイス名。/dev/md/dsk/dnum の形式で表示されます。
 - Veritas ボリュームマネージャのボリューム名。/dev/vx/dsk/volume_name の形式で表示されます。
 - クリティカルファイルシステムの選択領域は、クリティカルファイルシステムを選択するまで空白のままです。/usr、/var、/opt などのクリティカルファイルシステムは、分割可能です。ルート (/) ファイルシステムとマージすることもできます。
 - /export などの共有可能なファイルシステム (スワップ) は「Device」フィールドに表示されます。これらのファイルシステムは、ソースブート環境とターゲットブート環境の両方に同じマウントポイントを持っています。スワップはデフォルトで共有されますが、スワップスライスの分割とマージ (追加と削除) も行うことができます。
クリティカルファイルシステムと共有可能なファイルシステムの概要は、234 ページの「ブート環境の作成 (概要)」を参照してください。
- 「FS_Type」フィールドでは、ファイルシステムの種類を変更できます。ファイルシステムの種類は、次のいずれかになります。

- vxfs - Veritas ファイルシステムを示す
- swap - スワップファイルシステムを示す
- ufs - UFS ファイルシステムを示す

4. (省略可能) 以下の作業は、必要に応じて行うことができます。

- 出力画面情報を ASCII ファイルに出力するには F5 を押します。
- ファイルシステムのリストをスクロールするには、Control-X を押します。
この操作で、アクティブブート環境のファイルシステムと新しいブート環境のファイルシステムを切り替えてスクロールできるようになります。
- 「Configuration」メニューを閉じるには、F6 を押します。
 - 「Configuration」メニューが表示されている場合は、変更が保存されず、ファイルシステムは修正されません。
 - 「Configuration」サブメニューが表示されている場合は、「Configuration」メニューに戻ります。

5. F2 を押して、利用できるスライスを選択します。

「Choices」メニューでは、カーソルが置かれているフィールドに、そのシステム上で利用できるスライスが表示されます。表示されるのは「Slice」フィールドと「file system FS_Type」フィールドです。

- a. 矢印キーを使用してフィールド内にカーソルを置き、スライスまたはファイルシステムの種類を選択します。
 - 「Slice」フィールドにカーソルを置くと、すべての空きスライスが表示されます。ルート (/) の場合、「Choices」に表示されるのはルート (/) ファイルシステムの制限事項を満たす空きスライスだけです。251 ページの「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」を参照してください。
 - 「FS_Type」フィールドにカーソルを置くと、利用できるすべてのファイルシステムタイプが表示されます。
 - 現在のファイルシステムには、ボールド書体のスライスを選択できます。スライスのサイズは、ファイルシステムのサイズにアップグレード用に 30% を加えることによって概算できます。
 - ボールド書体ではないスライスは、そのファイルシステムをサポートするにはサイズが小さすぎることを意味します。ディスクスライスを作成し直す方法については、手順 6 を参照してください。

b. Return キーを押してスライスを選択します。

選択したスライスが「Slice」フィールドに表示されるか、あるいは「FS_Type」フィールド内でファイルシステムの種類が変化します。

6. (省略可能) 空きスライスが最小要件を満たしていない場合は、F4 を押して任意の空きスライスを分割し直してください。

「Solaris Live Upgrade Slice Configuration」メニューが表示されます。

新しいスライスを作成できるように、format コマンドが実行されます。画面の指示に従って新しいスライスを作成してください。format コマンドについての詳細

は、format のマニュアルページを参照してください。

矢印キーを使って、「Device」フィールドと「FS_Type」フィールド間を移動できます。デバイスを選択すると、「Size (Mbytes)」フィールドに値が自動的に表示されます。

a. デバイスを解放するには、**Control-D** を押します。
以上の操作でスライスが利用できるようになり、「Choices」メニューに表示されます。

b. **F3** を押して「**Configuration**」メニューに戻ります。

7. (省略可能) クリティカルファイルシステムを分割すると、そのファイルシステムは別々のマウントポイントに配置されます。ファイルシステムを分割するには、次の手順に従ってください。

(ファイルシステムのマージについては、手順 8 を参照してください。)

a. 分割するファイルシステムを選択します。

/usr、/var、/opt などのファイルシステムは、それらの親ディレクトリから分割できます。

注 – ブート環境のファイルシステムを作成する場合のルールは、Solaris オペレーティング環境のファイルシステムを作成する場合と同じです。Solaris Live Upgrade では、クリティカルな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 コマンドを用いて、ルート (/) と /kernel を別々のファイルシステムに作成することができますが、このようにルート (/) を分割するのは誤りです。

b. **F8** を押します。

c. 新しいブート環境のファイルシステム名を入力します。

例:

```
Enter the directory that will be a separate file system on the new boot environment: /opt
新しいファイルシステムが検証されると、画面に新しい行が追加されます。
```

d. **F3** を押して「**Configuration**」メニューに戻ります。

「Configuration」メニューが表示されます。

8. (省略可能) マージを行うと、ファイルシステムは同じマウントポイントに配置されます。ファイルシステムをその親ディレクトリにマージするには、次の操作を行います。

(ファイルシステムの分割については、手順 7 を参照してください。)

a. マージするファイルシステムを選択します。

/usr、/var、/opt などのファイルシステムをそれらの親ディレクトリにマージできます。

- b. **F9** を押します。
次の例のように、結合されるファイルシステムが表示されます。

```
/opt will be merged into /.
```
 - c. Return キーを押します。
 - d. **F3** を押して「**Configuration**」メニューに戻ります。
「**Configuration**」メニューが表示されます。
9. (省略可能) スワップスライスを追加するか削除するかを決定します。
- スワップスライスを分割して新しいスライスに配置したい場合は、手順 10 に進みます。
 - スワップスライスを削除したい場合は、手順 11 に進みます。
10. (省略可能) スワップスライスを分割するには、次の手順に従います。
- a. 「**Device**」フィールドで、分割したいスワップスライスを選択します。
 - b. **F8** を押します。
 - c. プロンプトに対して、次のコマンドを入力します。

```
Enter the directory that will be a separate filesystem on the new BE: swap
```
 - d. **F2 (Choice)** を押します。
「**Choice**」メニューに、スワップに利用できるスライスが表示されます。
 - e. スワップを配置するスライスを選択します。
そのスライスが「**Device**」フィールドに表示され、スワップの新しいスライスとなります。
11. (省略可能) スワップスライスを削除するには、次の手順に従います。
- a. 「**Device**」フィールドで、削除するスワップスライスを選択します。
 - b. **F9** を押します。
 - c. プロンプトが表示されたら「**y**」と入力します。

```
Slice /dev/dsk/c0t4d0s0 will not be swap partition.  
Please confirm? [y, n]: y
```


このスワップスライスが削除されました。
12. 今すぐにブート環境を作成するか、後で作成するようにスケジュールするかを決定します。
- すぐに新しいブート環境を作成する場合は、**F3** を押します。

構成が保存され、構成画面が閉じます。このファイルシステムがコピーされ、ブート環境がブート可能になり、非アクティブブート環境が作成されます。

ブート環境の作成には、システム構成に応じて、1時間以上かかる場合があります。続いて、Solaris Live Upgrade メインメニューが表示されます。

- 後で作成されるようにスケジュールする場合は、次の例に示すように「y」と入力し、続いて開始時刻と電子メールアドレスを入力します。

```
Do you want to schedule the copy? y
Enter the time in 'at' format to schedule create: 8:15 PM
Enter the address to which the copy log should be mailed: someone@anywhere.com
```

処理の完了は、電子メールで通知されます。

時間の書式については、at (1) のマニュアルページを参照してください。

スケジュールできるのは一度に1つのジョブだけです。

作成が完了すると、非アクティブブート環境をアップグレードできるようになります。第33章を参照してください。

▼ ブート環境を初めて作成する (コマンド行インタフェース)

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クリティカルファイルシステムがある場合、このファイルシステムは、-m オプションで指定しなくても新しく作成されたファイルシステムの上位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したい場合は、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、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。

1. スーパーユーザーとしてログインします。

2. 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:fs_type [-m mountpoint:device:fs_type] -n BE_name
```

-A 'BE_description'	(省略可能) 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。このオプションが必要なのは、最初のブート環境を作成する場合だけです。lucreate を初めて実行する場合に -c を省略すると、アクティブブート環境の名前を入力するように求められます。最初のブート環境の作成後に -c オプションを使用すると、エラーメッセージが表示されます。
-m mountpoint:device:fs_type [-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、作成したいファイルシステムの数だけ使用します。 <ul style="list-style-type: none"> ■ <i>mountpoint</i> 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。 ■ <i>device</i> フィールドには、次のどちらかを指定できます。 <ul style="list-style-type: none"> ■ ディスクデバイスの名前。/dev/dsk/c numdnumtnums num の形式で表されます。 ■ Solaris ポリウムマネージャのメタデバイス名。/dev/md/dsk/dnum の形式で表されます。 ■ Veritas ポリウムマネージャのポリウム名。/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。 ■ <i>fs_type</i> フィールドには、次のいずれかを指定できます。 <ul style="list-style-type: none"> ■ <i>ufs</i> - UFS ファイルシステムを示します。 ■ <i>vxfs</i> - Veritas ファイルシステムを示します。 ■ <i>swap</i> - スワップファイルシステムを示します。スワップマウントポイントはハイフン (-) で表されます。
-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 32-1 ブート環境を作成する (コマンド行)

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。「Solaris 9 test Jan. 2001」という記述は、`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、自動的にソースである `first_disk` から共有されます。

```
# lucreate -A 'Solaris 9 test Jan 2001' -c first_disk
-m /:/dev/dsk/c0t4d0s0:ufs -m /usr:/dev/dsk/c0t4d0s3:ufs \
-n second_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 33 章を参照してください。

▼ ブート環境を作成しファイルシステムをマージする (コマンド行インタフェース)

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。

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

```
# lucreate [-A 'BE_description'] -m mountpoint:device:fs_type \
[-m mountpoint:device:fs_type] -m mountpoint:merged:fs_type -n BE_name
```

`-A BE_description` (省略可能) 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。

<p><code>-m mountpoint:device:fs_type [-m...]</code></p>	<p>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。 <code>-m</code> 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、作成したいファイルシステムの数だけ使用します。</p> <ul style="list-style-type: none"> ■ <code>mountpoint</code> 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 <code>-</code> (ハイフン) を指定できます。 ■ <code>device</code> フィールドには、次のどちらかを指定できます。 <ul style="list-style-type: none"> ■ ディスクデバイスの名前。 <code>/dev/dsk/c numdnumtnums num</code> の形式で表されます。 ■ Solaris ポリリュームマネージャのメタデバイス名。 <code>/dev/md/dsk/dnum</code> の形式で表されます。 ■ Veritas ポリリュームマネージャのポリリューム名。 <code>/dev/vx/dsk/volume_name</code> の形式で表されます。 ■ キーワード <code>merged</code>。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。 ■ <code>fs_type</code> フィールドには、次のいずれかを指定できます。 <ul style="list-style-type: none"> ■ <code>ufs</code> - UFS ファイルシステムを示します。 ■ <code>vxfs</code> - Veritas ファイルシステムを示します。 ■ <code>swap</code> - スワップファイルシステムを示します。スワップマウントポイントはハイフン <code>-</code> で表されます。
<p><code>-n BE_name</code></p>	<p>作成するブート環境の名前。 <code>BE_name</code> 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>

例 32-2 ブート環境を作成しファイルシステムをマージする (コマンド行インタフェース)

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。「Solaris 9 test Jan. 2001」という記述は、`second_disk` に対応しています。

```
# lucreate -A 'Solaris 9 test Jan 2001' -c first_disk \
-m /:/dev/dsk/c0t4d0s0:ufs -m /usr:/dev/dsk/c0t4d0s1:ufs \
-m /usr/opt:merged:ufs -n second_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 33 章を参照してください。

▼ ブート環境を作成しファイルシステムを分割する (コマンド行インタフェース)

注 – ブート環境のファイルシステムを作成する場合のルールは、Solaris オペレーティング環境のファイルシステムを作成する場合と同じです。Solaris Live Upgrade では、クリティカルな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 コマンドを用いて、ルート (/) と /kernel を別々のファイルシステムに作成することができますが、このようにルート (/) を分割するのは誤りです。

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。

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

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

-A 'BE_description' (省略可能) 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:fs_type [-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、作成したい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、次のどちらかを指定できます。
 - ディスクデバイスの名前。 `/dev/dsk/c numdnumtnums num` の形式で表されます。
 - Solaris ポリリュームマネージャのメタデバイス名。 `/dev/md/dsk/dnum` の形式で表されます。
 - Veritas ポリリュームマネージャのポリリューム名。 `/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_type` 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` - スワップファイルシステムを示します。スワップマウントポイントはハイフン `-` で表されます。

`-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 32-3 ブート環境を作成しファイルシステムを分割する (コマンド行インタフェース)

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` がすべてルート (`/`) にある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
```

「Solaris 9 test Jan. 2001」という記述は、ブート環境名 `second_disk` に対応していません。

```
# lucreate -A 'Solaris 9 test Jan 2001' -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
```

例 32-3 ブート環境を作成しファイルシステムを分割する (コマンド行インタフェース) (続き)

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 33 章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 (コマンド行インタフェース)

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したい場合は、-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、スワップスライスは使用し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。

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

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

-A 'BE_description' (省略可能) 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。

<code>-m mountpoint:device:fs_type [-m...]</code>	<p>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、作成したいファイルシステムの数だけ使います。</p> <ul style="list-style-type: none"> ■ <code>mountpoint</code> 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。 ■ <code>device</code> フィールドには、次のどちらかを指定できます。 <ul style="list-style-type: none"> ■ ディスクデバイスの名前。/dev/dsk/c numdnumt numnum の形式で表されます。 ■ Solaris ポリリュームマネージャのメタデバイス名。/dev/md/dsk/dnum の形式で表されます。 ■ Veritas ポリリュームマネージャのポリリューム名。/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。 ■ <code>fs_type</code> フィールドには、次のいずれかを指定できます。 <ul style="list-style-type: none"> ■ <code>ufs</code> - UFS ファイルシステムを示します。 ■ <code>vxfs</code> - Veritas ファイルシステムを示します。 ■ <code>swap</code> - スワップファイルシステムを示します。スワップマウントポイントはハイフン (-) で表されます。
<code>-n BE_name</code>	<p>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。</p>

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。

例 32-4 ブート環境を作成しスワップを再構成する (コマンド行インタフェース)

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。「Solaris 9 test Jan. 2001」という記述は、ブート環境名 second_disk に対応しています。

```
# lucreate -A 'Solaris 9 test Jan 2001' -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。スワップスライスのリストが長い場合は、-M オプションを使用すると便利です。273 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 (コマンド行インタフェース)」を参照してください。

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 33 章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 (コマンド行インタフェース)

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、スワップスライスが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、スワップスライスが他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` ファイルにはデバイスとスライスが挙げられています。

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

2. 次のコマンドを入力します。

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

`-A 'BE_description'` (省略可能) ブート環境名 (BE_name) の説明を記述します。記述の長さ、使用できる文字に制限はありません。

<p><code>-m mountpoint:device:fs_type [-m...]</code></p>	<p>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。 <code>-m</code> 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、作成したいファイルシステムの数だけ使用します。</p> <ul style="list-style-type: none"> ■ <code>mountpoint</code> 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 <code>-</code> (ハイフン) を指定できます。 ■ <code>device</code> フィールドには、次のどちらかを指定できます。 <ul style="list-style-type: none"> ■ ディスクデバイスの名前。 <code>/dev/dsk/c numdnumtnums num</code> の形式で表されます。 ■ Solaris ポリリュームマネージャのメタデバイス名。 <code>/dev/md/dsk/dnum</code> の形式で表されます。 ■ Veritas ポリリュームマネージャのポリリューム名。 <code>/dev/vx/dsk/volume_name</code> の形式で表されます。 ■ キーワード <code>merged</code>。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。 ■ <code>fs_type</code> フィールドには、次のいずれかを指定できます。 <ul style="list-style-type: none"> ■ <code>ufs</code> - UFS ファイルシステムを示します。 ■ <code>vxfs</code> - Veritas ファイルシステムを示します。 ■ <code>swap</code> - スワップファイルシステムを示します。スワップマウントポイントはハイフン <code>-</code> で表されます。
<p><code>-M slice_list</code></p>	<p>ファイル <code>slice_list</code> 中には、<code>-m</code> 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、<code>-m</code> 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 (<code>#</code>) で始まるコメント行は無視されます。<code>-M</code> 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。<code>-m</code> オプションと <code>-M</code> オプションは結合できます。たとえば、<code>slice_list</code> にスワップスライスを記録しておき、<code>-m</code> を使用して、ルート (<code>/</code>) スライスと <code>/usr</code> スライスを指定できます。</p> <p><code>-m</code> オプションと <code>-M</code> 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、<code>lucreate</code> 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。</p>
<p><code>-n BE_name</code></p>	<p>作成するブート環境の名前。<code>BE_name</code> は、一意となるように指定する必要があります。</p>

例 32-5 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 (コマンド行インタフェース)

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。「Solaris 9 test Jan. 2001」という記述は、`second_disk` に対応しています。

```
# lucreate -A 'Solaris 9 test Jan 2001' -c first_disk \
-m /:/dev/dsk/c02t4d0s0:ufs -m /usr:/dev/dsk/c02t4d0s1:ufs \
-M /etc/lu/swapslices -n second_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 33 章を参照してください。

▼ 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 (コマンド行インタフェース)

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したい場合は、`-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、両方の環境に適用されます。

1. スーパーユーザーとしてログインします。
2. ブート環境を作成します。

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

<code>-A 'BE_description'</code>	(省略可能) ブート環境名 (BE_name) の説明を記述します。記述の長さ、使用できる文字に制限はありません。
<code>-m mountpoint:device:fs_type [-m...]</code>	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、作成したいファイルシステムの数だけ使用します。 <ul style="list-style-type: none">■ <code>mountpoint</code> 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。■ <code>device</code> フィールドには、次のどちらかを指定できます。<ul style="list-style-type: none">■ ディスクデバイスの名前。/dev/dsk/c numdnumtnums num の形式で表されます。■ Solaris ポリウムマネージャのメタデバイス名。/dev/md/dsk/dnum の形式で表されます。■ Veritas ポリウムマネージャのポリウム名。/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。■ <code>fs_type</code> フィールドには、次のいずれかを指定できます。<ul style="list-style-type: none">■ <code>ufs</code> - UFS ファイルシステムを示します。■ <code>vxfs</code> - Veritas ファイルシステムを示します。■ <code>swap</code> - スワップファイルシステムを示します。スワップマウントポイントはハイフン (-) で表されます。
<code>-n BE_name</code>	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。

例 32-6 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 (コマンド行インタフェース)

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。「Solaris 9 test Jan. 2001」という記述は、ブート環境名 second_disk に対応しています。

```
# lucreate -A 'Solaris 9 test Jan 2001' -c first_disk \  
-m /:/dev/dsk/c0t4d0s0:ufs -m /usr:/dev/dsk/c0t4d0s3:ufs \  
-m /home:/dev/dsk/c0t4d0s4:ufs -n second_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 33 章を参照してください。

▼ 別々のソースから単一のブート環境を作成 (コマンド行インタフェース)

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最後にアクティブだったブート環境に戻ります。

1. スーパーユーザーとしてログインします。
2. ブート環境を作成します。

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

-A 'BE_description' (省略可能) 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:fs_type [-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、作成したい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、次のどちらかを指定できます。
 - ディスクデバイスの名前。 `/dev/dsk/c numdnumt num` の形式で表されます。
 - Solaris ボリュームマネージャのメタデバイス名。 `/dev/md/dsk/dnum` の形式で表されます。
 - Veritas ボリュームマネージャのボリューム名。 `/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_type` 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` - スワップファイルシステムを示します。スワップマウントポイントはハイフン (`-`) で表されます。
- `-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 32-7 別々のソースから単一のブート環境を作成 (Command-Line Interface)

この例では、ソースブート環境 `third_disk` 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。「Solaris 9 test Jan. 2001」という記述は、ブート環境名 `second_disk` に対応しています。

```
# lucreate -A 'Solaris 9 test Jan 2001' -s third_disk \  
-m /:/dev/dsk/c0t4d0s0:ufs -n second_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 33 章を参照してください。

▼ フラッシュアーカイブ用ブート環境の作成 (コマンド行インタフェース)

`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、フラッシュアーカイブのインストールに利用できる空のブート環境を非常に短時間で作成できます。

1. スーパーユーザーとしてログインします。

2. 空のブート環境を作成します。

```
# lucreate -A 'BE_name' -s - \  
-m mountpoint:device:fs_type -n BE_name
```

-A 'BE_description' (省略可能) 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:fs_type [-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、作成したい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、次のどちらかを指定できます。
 - ディスクデバイスの名前。/dev/dsk/c numdnumtnums num の形式で表されます。
 - Solaris ポリウムマネージャのメタデバイス名。/dev/md/dsk/dnum の形式で表されます。
 - Veritas ポリウムマネージャのポリウム名。/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_type** 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** - スワップファイルシステムを示します。スワップマウントポイントはハイフン (-) で表されます。

-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 32-8 フラッシュアーカイブ用ブート環境の作成 (コマンド行インタフェース)

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。「Solaris 9 test Jan. 2001」という記述は、ブート環境名 second_disk に対応しています。

```
# lucreate -A 'Solaris 9 test Jan 2001' -s - \  
-m /:/dev/dsk/c0t4d0s0: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 33 章を参照してください。

第 33 章

Solaris Live Upgrade を用いたアップグレード

この章では、Solaris 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。また、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も説明します。この章の内容は次のとおりです。

- 280 ページの「作業マップ: ブート環境のアップグレード」
- 280 ページの「ブート環境のアップグレード」
- 286 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」
- 291 ページの「ブート環境のアクティブ化」
- 294 ページの「問題の解決: 元のブート環境へのフォールバック (コマンド行インタフェース)」

Solaris Live Upgrade は、メニューを介して使用することもコマンド行インタフェースで使用することもできます。以下の説明では、両方のインタフェースについて手順を説明しています。これらの手順では、Solaris Live Upgrade の使用に関する詳しい説明は省略しています。コマンドについての詳細は第 36 章を、コマンド行インタフェースについての詳細は各マニュアルページを参照してください。

作業マップ: ブート環境のアップグレード

表 33-1 作業マップ: Solaris Live Upgrade によるアップグレード

作業	説明	参照先
ブート環境のアップグレードまたはフラッシュアーカイブのインストール	<ul style="list-style-type: none">■ オペレーティング環境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へ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">■ 280 ページの「ブート環境のアップグレード」■ 286 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」
非アクティブブート環境のアクティブ化	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。	291 ページの「ブート環境のアクティブ化」
(任意) アクティブ化で問題が発生した場合のフォールバック	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。	294 ページの「問題の解決: 元のブート環境へのフォールバック (コマンド行インタフェース)」

ブート環境のアップグレード

「Upgrade」メニューまたは `luupgrade` コマンドを使用してブート環境をアップグレードします。この節では、以下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。

- ネットワークファイルシステム (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)

注 - インストールに複数の CD が必要な場合は、コマンド行インタフェースを使用してアップグレードする必要があります。284 ページの「複数の CD を使用してオペレーティングシステムイメージをアップグレードする (コマンド行インタフェース)」を参照してください。

最新のオペレーティング環境へアップグレードする間、アクティブブート環境への影響はありません。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。

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。286 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。

メタデバイスやボリュームのアップグレード

新しいブート環境にアップグレードを行ったり、フラッシュアーカイブを使ってインストールを行う場合、そのデバイスは物理スライスでなければなりません。Solaris ボリュームマネージャメタデバイスか Veritas ファイルシステム (VxFS) ボリュームにマウントされているファイルシステムがそのブート環境にあると、アップグレードまたはフラッシュアーカイブのインストールは失敗します。そのようなブート環境でアップグレードやフラッシュアーカイブのインストールを行う場合は、すべてのスライスが物理ディスクスライスになるようにブート環境を手動で変更する必要があります。lucreate コマンドを使って別のブート環境を作成することも、メタデバイスやボリュームを削除することもできます。

lucreate コマンドを使って別のブート環境を作成する場合は、ブート環境のコピーを物理ディスクスライスに格納します。たとえば、現在のブート環境のディスク構成が次のようになっているとします。

/	/dev/md/dsk/d10 にマウントされている
/usr	/dev/md/dsk/d20 にマウントされている
/var	/dev/dsk/c0t3d0s0 にマウントされている
空きスライス	c0t4d0s0
空きスライス	c0t4d0s3
空きスライス	c0t4d0s4

この場合、次の lucreate コマンドを使ってブート環境を物理ディスクスライスにコピーします。これで、ブート環境は、アップグレードやフラッシュアーカイブを用いたインストールを行える状態になります。この例では、現在のブート環境の名前が currentBE、新しいブート環境の名前が nextBE であるとしています。

```
# lucreate -s currentBE -n nextBE -m /:/dev/dsk/c0t4d0s0:ufs \  
-m /usr:/dev/dsk/c0t4d0s3:ufs -m /var:/dev/dsk/c0t4d0s4:ufs
```

これで、新しいブート環境に対してアップグレードやフラッシュアーカイブを用いたインストールを行い、それをアクティブ化し、さらに手動で再ミラー化またはカプセル化することができます。

メタデバイスやボリュームを手動で削除する場合は、次のようにします。

- ブート環境を手動でマウントします。
- `/etc/vfstab` ファイルを編集して、すべてのメタデバイスやボリュームを物理ディスクスライスに変更します。
- Solaris ボリュームマネージャまたは Veritas ソフトウェアを使ってメタデバイスまたはボリュームを削除します。
- ブート環境をマウント解除します。

これで、ブート環境は、アップグレードやフラッシュアーカイブを用いたインストールを行える状態になります。

Solaris Live Upgrade によるパッケージやパッチの管理

Solaris Live Upgrade を使ってパッチやパッケージをシステムに追加することができます。マシンアップグレードの場合と同じように、Live Upgrade を使ってパッチをマシンに追加すると、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 コマンドを使えば、パッチやパッケージをブート環境に直接追加できます。あるいは、パッチやパッケージをマスターシステムに適用し、そのシステムのフラッシュアーカイブを Live Upgrade でブート環境に適用することもできます。

パッチをブート環境に直接追加する場合は、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。パッケージをブート環境に追加する場合は、luupgrade コマンドに `-p` オプションを指定します。詳細は、luupgrade (1M) のマニュアルページを参照してください。

あるいは、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を含むフラッシュアーカイブを Solaris Live Upgrade を使用してインストールすることもできます。この完全なブート環境または単一のリファレンスシステムをマスターシステムと呼びます。フラッシュアーカイブを作成するプロセスは、マスターシステムの作成から始まります。マスターシステムを作成したら、インストールしたいパッチやパッケージを追加します。そして、マスターシステムのフラッシュアーカイブを作成します。次に Solaris Live Upgrade を使ってこのアーカイブを新しいブート環境にインストールします。ブート環境は、必要に応じて何回でもコピーしたり、変更したり、配布したりすることができます。フラッシュアーカイブの作成方法については、第 18 章を参照してください。Solaris Live Upgrade を使ってフラッシュアーカイブをインストールする方法については、286 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。



注意 – Solaris Live Upgrade でパッケージやパッチのアップグレードや追加、削除を行うためには、パッケージやパッチが SVR4 パッケージ化ガイドラインに準拠していなければなりません。サンのパッケージはこのガイドラインに準拠していますが、サードパーティベンダーのパッケージがこれに準拠しているとは限りません。パッケージがガイドラインに準拠していないと、アップグレード時にパッケージ追加ソフトウェアに問題が発生したり、場合によっては、アクティブなブート環境が変更されたりすることがあります。

Solaris Live Upgrade によるパッケージの追加や削除については、luupgrade (1M) のマニュアルページを参照してください。パッケージ化の要件については、付録 C を参照してください。

▼ 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する (キャラクタインタフェース)

この手順でアップグレードを行う場合は、1つの CD、DVD、または結合されたインストールイメージを使用する必要があります。複数の CD が必要なインストールでは 284 ページの「複数の CD を使用してオペレーティングシステムイメージをアップグレードする (コマンド行インタフェース)」の手順を使用してください。

1. **Solaris Live Upgrade** のメインメニューから「**Upgrade**」を選択します。「Upgrade」メニューが表示されます。
2. 新しいブート環境の名前を入力します。
3. **Solaris** インストールイメージが置かれている場所のパスを入力します。

インストールメディアの種類	説明
ネットワークファイルシステム	インストールイメージが置かれているネットワークファイルシステムのパスを指定します。
ローカルファイル	インストールイメージが置かれているローカルファイルシステムのパスを指定します。
ローカルテープ	インストールイメージが置かれているローカルテープデバイスとテープ上の位置を指定します。
ローカルデバイス、DVD、または CD	ローカルデバイスと、インストールイメージのパスを指定します。

- DVD または 1 つの CD を使用する場合は、次の例のように、そのディスクへのパスを入力します。

```
Package Media: /cdrom/solaris9/s0
```

- 1つに結合されたイメージがネットワーク上に存在する場合は、次の例のようにそのネットワークファイルシステムのパスを入力します。

Package Media: `/net/installmachine/export/solaris9/os_image`

4. **F3** を押してアップグレードします。
5. アップグレードが完了すると、メインメニューが表示されます。

▼ 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する (コマンド行インタフェース)

この手順でアップグレードを行う場合は、1つの CD、DVD、または結合されたインストールイメージを使用する必要があります。複数の CD が必要なインストールでは、284 ページの「複数の CD を使用してオペレーティングシステムイメージをアップグレードする (コマンド行インタフェース)」を参照してください。

1. スーパーユーザーとしてログインします。
2. 次のように入力して、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
```

<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。

例 33-1 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する (コマンド行インタフェース)

この例では、`second_disk` ブート環境をアップグレードします。

```
# luupgrade -u -n second_disk \  
-s /net/installmachine/export/solaris9/OS_image
```

▼ 複数の CD を使用してオペレーティングシステムイメージをアップグレードする (コマンド行インタフェース)

オペレーティングシステムイメージが複数の CD に入っている場合は、このアップグレード方法を使用してください。

1. スーパーユーザーとしてログインします。
2. 次のように入力して、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	アップグレードする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。

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. この手順は前述のものと同じですが、-u オプションではなく -i オプションを使用します。メニューまたはテキストモードで、2 枚目の **CD** 上のインストーラを実行するように選択してください。

- 次のコマンドは、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します。

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

- 次のコマンドは、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。

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

-i	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	オペレーティングシステムイメージを含むディレクトリのパス名を指定します。
-O '-nodisplay -noconsole'	(任意) 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。

以上の手順で、ブート環境をアクティブにする準備が整います。291 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 33-2 複数の CD によりオペレーティングシステムイメージをアップグレードする (コマンド行インタフェース)

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 Solaris 9 SOFTWARE 1 of 2 と Solaris 9 SOFTWARE 2 of 2 CD に入っています。-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

例 33-2 複数の CD によりオペレーティングシステムイメージをアップグレードする (コマンド行インタフェース) (続き)

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。したがって、入力情報のプロンプトは表示されません。このインタフェースを表示する場合は、これらのオプションを省略します。

Solaris 9 SOFTWARE 1 of 2 CD を挿入し、次のように入力します。

```
# luupgrade -u -n second_disk -s /dev/cdrom/cdrom0/s0
```

Solaris 9 SOFTWARE 2 of 2 CD を挿入し、次のように入力します。

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

注 - 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。

ブート環境へのフラッシュアーカイブのインストール

この節では、以下の場所に格納されているフラッシュアーカイブを、Solaris Live Upgrade を使用してインストールする手順を説明します。

- HTTP サーバー
- FTP サーバー - このパスは、コマンド行だけから使用します。
- ネットワークファイルシステム (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)

注 - アーカイブ記憶域に関連するパスの構文例については、204 ページの「archive_location プロファイルキーワード」を参照してください。

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存在す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。

フラッシュ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ておく必要があります。フラッシュの詳細は、第 16 章を参照してください。

メタデバイスとボリュームのアップグレード

新しいブート環境にアップグレードを行ったりフラッシュアーカイブを使ってインストールを行う場合、そのデバイスは物理スライスでなければなりません。Solaris ボリュームマネージャメタデバイスか Veritas ファイルシステム (VxFS) ボリュームにマウントされているファイルシステムがそのブート環境にあると、アップグレードまたはフラッシュアーカイブのインストールは失敗します。そのようなブート環境でアップグレードやフラッシュアーカイブのインストールを行う場合は、すべてのスライスが物理ディスクスライスになるようにブート環境を手動で変更する必要があります。lucreate コマンドを使って別のブート環境を作成することも、メタデバイスやボリュームを削除することもできます。

lucreate コマンドを使って別のブート環境を作成する場合は、ブート環境のコピーを物理ディスクスライスに格納します。たとえば、現在のブート環境のディスク構成が次のようになっているとします。

```
/                /dev/md/dsk/d10 にマウントされている
/usr             /dev/md/dsk/d20 にマウントされている
/var            /dev/dsk/c0t3d0s0 にマウントされている
空きスライス   c0t4d0s0
空きスライス   c0t4d0s3
空きスライス   c0t4d0s4
```

この場合、次の lucreate コマンドを使ってブート環境を物理ディスクスライスにコピーします。これで、ブート環境は、アップグレードやフラッシュアーカイブを用いたインストールを行える状態になります。この例では、現在のブート環境の名前が currentBE、新しいブート環境の名前が nextBE であるとしています。

```
# lucreate -s currentBE -n nextBE -m /:/dev/dsk/c0t4d0s0:ufs \
-m /usr:/dev/dsk/c0t4d0s3:ufs -m /var:/dev/dsk/c0t4d0s4:ufs
```

これで、新しいブート環境に対してアップグレードやフラッシュアーカイブを用いたインストールを行い、それをアクティブ化し、さらに手動で再ミラー化またはカプセル化することができます。

メタデバイスやボリュームを手動で削除する場合は、次のようにします。

- ブート環境を手動でマウントします。
- /etc/vfstab ファイルを編集して、すべてのメタデバイスやボリュームを物理ディスクスライスに変更します。

- Solaris ボリュームマネージャまたは Veritas ソフトウェアを使ってメタデバイスまたはボリュームを削除します。
- ブート環境をマウント解除します。

これで、ブート環境は、アップグレードやフラッシュアーカイブを用いたインストールを行える状態になります。

▼ ブート環境へのフラッシュアーカイブのインストール (キャラクタインタフェース)

1. **Solaris Live Upgrade** のメインメニューから「**Flash**」を選択します。
「Flash an Inactive Boot Environment」メニューが表示されます。
2. フラッシュアーカイブをインストールするブート環境の名前と、インストールメディアの場所を入力します。

```
Name of Boot Environment: solaris_9  
Package media: /net/install-svr/export/s9/latest
```

3. アーカイブを追加するために **F1** を押します。
「Archive Selection」サブメニューに、空のリスト、単一のアーカイブの名前、または複数のアーカイブの一覧が表示されます。複数のアーカイブが表示される場合は、アーカイブが階層化されています。階層化されたアーカイブの説明については、112 ページの「階層化された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。
リストにアーカイブを追加することも、リストからアーカイブを削除することもできます。次に、空のリストの例を示します。

```
Location - Retrieval Method  
<No Archives added> - Select ADD to add archives
```

- 変更を加えずにリストをインストールする場合は、手順 4 へ進みます。
 - フラッシュアーカイブのリストが空であるか、あるいはインストールしたくないアーカイブが含まれている場合は、手順 a へ進みます。
- a. アーカイブを追加または削除することができます。
 - リストにアーカイブを追加する場合は **F1** を押します。手順 b へ進みます。

注 – 複数のアーカイブが表示される場合は、アーカイブが階層化されていません。詳細は、112 ページの「階層化された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。

「Select Retrieval Method」サブメニューが表示されます。

HTTP
NFS
Local File
Local Tape
Local Device

- リストからアーカイブを削除する場合は F2 を押します。手順 e へ進みます。

- b. 「**Select Retrieval Method**」メニューで、フラッシュアーカイブの場所を選択します。

選択された媒体	プロンプト
HTTP	フラッシュアーカイブへのアクセスに必要な URL とプロキシ情報を指定してください。
NFS	フラッシュアーカイブを含むネットワークファイルシステムのパスを指定してください。アーカイブのファイル名も指定できます。
ローカルファイル	フラッシュアーカイブを含むローカルファイルシステムのパスを指定してください。
ローカルテープ	フラッシュアーカイブを含むローカルテープデバイスとテープ上の位置を指定してください。
ローカルデバイス	フラッシュアーカイブを含むローカルデバイスとパス、ファイルシステムのタイプを指定してください。

次のような「Retrieval」サブメニューが表示されます。表示は選択されたメディアによって異なります。

NFS Location:

- c. 次の例のように、アーカイブのパスを入力します。

NFS Location: **host:/path/to archive.flar**

- d. **F3** を押してリストにアーカイブを追加します。

- e. インストールしたいアーカイブがリストに含まれた時点で **F6** を押して終了します。

4. **F3** を押してアーカイブ (単一のアーカイブまたは階層化された複数のアーカイブ) をインストールします。

フラッシュアーカイブがブート環境にインストールされます。ブート環境上のファイルは、共有可能ファイルを除きすべて上書きされます。

以上の手順で、ブート環境をアクティブにする準備が整います。詳細は、291 ページの「ブート環境のアクティブ化 (キャラクタインタフェース)」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 (コマンド行インタフェース)

1. スーパーユーザーとしてログインします。

2. 次のコマンドを入力します。

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

オプションとして、-a、-j、または -J のいずれかを指定する必要があります。

-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を含む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'	フラッシュインストール用に構成された JumpStart プロファイル内のエントリです。JumpStart ソフトウェアの詳細は、pfinstall (1M) のマニュアルページと第 21 章を参照してください。-a、-j、または -J のいずれかを指定する必要があります。
-j profile_path	フラッシュインストール用に構成された JumpStart プロファイルのパスです。JumpStart ソフトウェアの詳細は、pfinstall (1M) のマニュアルページと第 21 章を参照してください。-a、-j、または -J のいずれかを指定する必要があります。
-a archive	フラッシュアーカイブのパス (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)。-a、-j、または -J のいずれかを指定する必要があります。

例 33-3 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 (コマンド行インタフェース)

この例では、second_disk ブート環境にフラッシュアーカイブをインストールしています。-J オプションは、アーカイブを取り出すために使用されています。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。

```
# 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。詳細は、294 ページの「問題の解決: 元のブート環境へのフォールバック (コマンド行インタフェース)」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、そのブート環境が以下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」でなければなりません。ステータスを確認するには、300 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。
- 現在のブート環境とは別のブート環境をアクティブにする場合は、`lumount (1M)` または `mount (1M)` 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。
- 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。304 ページの「ブート環境の比較」を参照してください。

注 - スワップを再構成したい場合は、非アクティブブート環境をブートする前に行ってください。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。スワップを再構成する場合は、「新しいブート環境を作成する」、手順 9、または 271 ページの「ブート環境を作成しスワップを再構成する (コマンド行インタフェース)」を参照してください。

▼ ブート環境のアクティブ化 (キャラクタインタフェース)

新しく作成したブート環境で初めてブートする時に、Solaris Live Upgrade 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。アクティブなブート環境が、新しく作成したブート環境の元になったブート環境であるとは限りません。ここでいう「同期」は、前にアクティブであったブート環境の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を意味します。ただし、「Do you want to force a Live Upgrade sync operations」という質問に対し **yes** を入力しない限り、この最初のブート時に同期化は行われません。

同期化を実行すると、Solaris Live Upgrade は、同期をとるファイルの間に矛盾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、矛盾が検出されると、警告が出されます。この場合、ファイルの同期化は行われません。このような場合でも、アクティブ化は正常に終了することがあります。このような矛盾は、次の場合に発生することがあります。

- 現在のブート環境を新しいオペレーティングシステムにアップグレードする
- 非アクティブなブート環境を新しいオペレーティングシステムにアップグレードし、元のブート環境にあるファイルを変更する
- 両方のブート環境のシステムファイル (/etc/passwd など) を変更する

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 Solaris 9 ソフトウェアを実行しているとします。この場合、Solaris 2.6 リリースをブートし、同期化を実行すると、2.6 リリースのファイルは変更されることがあります。ファイルはオペレーティング環境のリリースに依存しているため、Solaris 2.6 リリースのブートは失敗することがあります。Solaris 9 のファイルと Solaris 2.6 のファイルは互換性があるとは限らないからです。

1. **Solaris Live Upgrade** のメインメニューで「**Activate**」を選択します。

2. アクティブにするブート環境の名前を入力します。

```
Name of Boot Environment: solaris_9
Do you want to force a Live Upgrade sync operations: no
```

3. ファイルの同期化を実行することも、実行せずに処理を続けることもできます。

- **Return** キーを押して継続します。

ブート環境の最初のブートでは、ファイルの同期化が自動的に行われます。

- ファイルの同期化を実行できますが、この機能は慎重に使用してください。同期化するファイルに、それぞれのブート環境のオペレーティングシステムが対応していなければなりません。ファイルの同期化を実行する場合は、次のように入力します。

```
Do you want to force a Live Upgrade sync operations: yes
```

4. **F3** を押して、アクティブ化の処理を開始します。

5. **Return** キーを押して継続します。

新しいブート環境は、次のリブート時にアクティブになります。

6. 非アクティブブート環境をリブートしてアクティブにします。

```
# init 6
```

▼ ブート環境をアクティブにする (コマンド行インタフェース)

1. スーパーユーザーとしてログインします。

2. 次のように入力して、ブート環境をアクティブにします。

```
# /usr/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。

3. 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。最後にアクティブであったブート環境が再びブートされます。

例 33-4 ブート環境をアクティブにする (コマンド行インタフェース)

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。

```
# /usr/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 (コマンド行インタフェース)

新しく作成したブート環境で初めてブートする時に、Solaris Live Upgrade 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。アクティブなブート環境が、新しく作成したブート環境の元になったブート環境であるとは限りません。ここでいう「同期」は、前にアクティブであったブート環境の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を意味します。この最初のブートの後で Solaris Live Upgrade がこの同期化を行うためには、`luactivate` に `-s` オプションを指定する必要があります。

`-s` オプションで同期化を実行すると、`luactivate` は、同期化に関連するファイルの間に矛盾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、矛盾が検出されると、警告が出されます。この場合、ファイルの同期化は行われません。このような場合でも、アクティブ化は正常に終了することがあります。このような矛盾は、次の場合に発生することがあります。

- 現在のブート環境を新しいオペレーティングシステムにアップグレードする
- 非アクティブなブート環境を新しいオペレーティングシステムにアップグレードし、元のブート環境にあるファイルを変更する
- 両方のブート環境のシステムファイル (`/etc/passwd` など) を変更する

以前のアクティブブート環境で発生した変更が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 Solaris 9 ソフトウェアを実行しているとした場合、Solaris 2.6 リリースをブートし、同期化を実行すると、2.6 リ

リリースのファイルは変更されることがあります。ファイルはオペレーティング環境のリリースに依存しているため、Solaris 2.6 リリースのブートは失敗することがあります。Solaris 9 のファイルと Solaris 2.6 のファイルは互換性があるとは限らないからです。

1. スーパーユーザーとしてログインします。
2. 次のように入力して、ブート環境をアクティブにします。

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

<code>BE_name</code>	アクティブにするブート環境の名前を指定します。
<code>-s</code>	最後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化します。 <code>-s</code> オプションは、ブート環境の最初のアクティブ化の後でファイルを同期化するときに使用します。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化が行われますが、それ以降のアクティブ化ではファイルの同期化は行われません。以前のアクティブブート環境で発生した変更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。

3. リポートします。

```
# init 6
```

例 33-5 ブート環境をアクティブにする (コマンド行インタフェース)

この例では、次のリポート時に `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。

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

問題の解決: 元のブート環境へのフォールバック (コマンド行インタフェース)

アップグレードの後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、次の手順のどれか (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フォールバックアップする必要があります。最初の手順では、`luactivate` を実行して元のブート環境に戻ります。この処理が失敗した場合は、2 つ目の手順によって DVD-ROM、CD-ROM、またはネットイメージからブートしてください。

▼ 新しいブート環境の作成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. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

```
# /usr/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。

3. 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。

▼ 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ため、元のブート環境をシングルユーザーモードでブートする場合は、この手順を使って元のブート環境にフォールバックします。
- 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、296 ページの「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フォールバックする」を参照してください。

1. **OK** プロンプトで、**Solaris 9 DVD**、**Solaris 9 INSTALLATION 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起動します。

```
OK boot cdrom -s
```

または

```
OK boot net -s
```

または

```
OK boot device_name -s
```

device_name システムをブートするデバイスの名前を指定します (たとえば、
/dev/dsk/c0t0d0s0)。

2. 次のコマンドを入力します。

```
# /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。

- このコマンドを実行した結果プロンプトが表示されない場合は、296 ページの「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フォールバックする」へ進みます。
- プロンプトが表示される場合は、次の手順に進みます。

3. プロンプトに対して、次のコマンドを入力します。

```
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。

▼ 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フォールバックする

DVD、CD、またはネットワークインストールイメージからブートする場合は、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。その後で、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起動されます。

1. **OK** プロンプトで、**Solaris 9 DVD**、**Solaris 9 INSTALLATION 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起動します。

```
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 (たとえば、`/dev/dsk/c0t0d0s0`)。

- 必要に応じて、フォールバックブート環境のルート (/) が置かれているファイルシステムの完全性を確認します。

```
# fsck device_name
```

device_name フォールバックしたい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。デバイス名の形式は、 /dev/dsk/c numtnum dnums num です。

- 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
```

device_name フォールバックしたい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。デバイス名の形式は、 /dev/dsk/c numtnum dnums num です。

- アクティブブート環境のルート (/) スライスから、次のように入力します。

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

前の稼働ブート環境がアクティブになり、結果が示されます。

- /mnt をマウント解除します。

```
# 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。

第 34 章

Solaris Live Upgrade ブート環境の管理

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、さまざまな管理作業について説明します。この章の内容は次のとおりです。

- 299 ページの「Solaris Live Upgrade 管理作業の概要」
- 300 ページの「すべてのブート環境のステータスの表示」
- 302 ページの「以前に構成されたブート環境の更新」
- 304 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」
- 304 ページの「ブート環境の比較」
- 306 ページの「非アクティブブート環境の削除」
- 307 ページの「アクティブブート環境の名前の表示」
- 308 ページの「ブート環境の名前の変更」
- 309 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」
- 312 ページの「ブート環境の構成の表示」

Solaris Live Upgrade 管理作業の概要

表 34-1 Solaris Live Upgrade 管理作業の概要

作業	説明	参照先
(省略可能) ステータスの表示	■ 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。	■ 300 ページの「すべてのブート環境のステータスの表示」

表 34-1 Solaris Live Upgrade 管理作業の概要 (続き)

作業	説明	参照先
	<ul style="list-style-type: none"> ■ 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"> ■ 304 ページの「ブート環境の比較」 ■ 307 ページの「アクティブブート環境の名前の表示」 ■ 312 ページの「ブート環境の構成の表示」
(省略可能) 非ブアクティブブート環境の更新	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。	302 ページの「以前に構成されたブート環境の更新」
(省略可能) その他の作業	<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。 ■ スケジュールされているジョブを取り消します。 	<ul style="list-style-type: none"> ■ 306 ページの「非アクティブブート環境の削除」 ■ 308 ページの「ブート環境の名前の変更」 ■ 309 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」 ■ 304 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」

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、「Status」メニューまたは `lustatus` コマンドを使用してください。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の状態 (作成またはコピーがスケジューリングされている、アクティブ、またはアップグレード中) を示します。ステータスが SCHEDULED の場合、Solaris 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。

▼ すべてのブート環境のステータスを表示する (キャラクタインタフェース)

- メインメニューから「**Status**」を選択します。
次のような表が表示されます。

BE_name	Complete	Active	ActiveOnReboot	CopyStatus
disk_a_s7	yes	yes	yes	-
disk_b_s7database	yes	no	no	COPYING
disk_b_s8	no	no	no	-

注 – この例では、disk_b_s8 は未完了の状態、disk_b_s7database は CopyStatus が SCHEDULED になっています。このためこれらの環境では、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。

▼ すべてのブート環境のステータスを表示する (コマンド行インタフェース)

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

```
# lustatus BE_name
```

BE_name

ステータスを表示する非アクティブブート環境の名前を指定します。*BE_name* を省略すると、lustatus によりシステム内のすべてのブート環境のステータスが表示されます。

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。

```
# lustatus
BE_name          Complete  Active  ActiveOnReboot  CopyStatus
-----
disk_a_s7        yes       yes     yes              -
```

disk_b_S7database	yes	no	no	COPYING
disk_b_S8	no	no	no	-

注 - この例では、disk_b_S8 は未完了の状態、disk_b_S7database は CopyStatus が SCHEDULED になっています。このためこれらの環境では、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。

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ると、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については、300 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。

コピー作業は後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処理を取り消す方法については、304 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。

▼ 以前に構成されたブート環境を更新する (キャラクタインタフェース)

1. メインメニューから「Copy」を選択します。
2. 更新する非アクティブブート環境の名前を入力します。

```
Name of Target Boot Environment: solaris8
```

3. コピー処理を継続するか、または後でコピーが実行されるようにスケジュールします。
 - コピーを継続するには、Return キーを押します。
以上の手順で、非アクティブブート環境が更新されます。
 - 後でコピーが実行されるようにスケジュールするには、「y」と入力し、時刻 (at コマンドの書式を使用) と、結果の送信先電子メールアドレスを指定します。

```
Do you want to schedule the copy? y
Enter the time in 'at' format to schedule copy: 8:15 PM
Enter the address to which the copy log should be mailed:
```

someone@anywhere.com

時間の書式については、at(1)のマニュアルページを参照してください。

以上の手順で、非アクティブブート環境が更新されます。

スケジュールされた処理を取り消す方法については、304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。

▼ 以前に構成されたブート環境を更新する(コマンド行インタフェース)

この手順では、以前に作成されたブート環境上の古いファイルを上書きしてソースファイルをコピーします。

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

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

-n <i>BE_name</i>	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 <i>source_BE</i>	(省略可能) 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 <i>time</i>	(省略可能) 指定されたブート環境上のファイルを指定された時刻に上書きするバッチジョブを設定します。時刻は、at(1)のマニュアルページに指定されている書式で入力します。
-m <i>email_address</i>	(省略可能) コマンドが完了した時点で、ここで指定する電子メールアドレスにlumakeの出力を送ります。 <i>email_address</i> 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 34-1 以前に構成されたブート環境を更新する(コマンド行インタフェース)

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に送信されます。

```
# 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処理を取り消す方法については、304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処理 (作成、アップグレード、コピー) は、その処理の開始前に取り消すことができます。GUI では、「Create a Boot Environment」、「Upgrade a Boot Environment」、または「Copy a Boot Environment」メニューを使用して、特定の時間に処理が実行されるようにスケジュールすることができます。CLI では、`lumake` コマンドを使用して、処理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処理 (作成/アップグレード/コピー) を取り消す (キャラクタインタフェース)

1. メインメニューから「**Cancel**」を選択します。
2. 取り消しが可能なブート環境の一覧を表示するには、**F2** を押します。
3. 取り消すブート環境を選択します。
これで、指定されている時刻に処理は実行されなくなります。

▼ スケジュールされた処理 (作成/アップグレード/コピー) を取り消す (コマンド行インタフェース)

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

```
# lucancel
```

これで、指定されている時刻に処理は実行されなくなります。

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、「Compare」メニューまたは `lucompare` コマンドを使用します。比較をするためには、非アクティブブート環境は完了状態で、コピー処理がスケジュールされていない必要があります。300 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。

指定するブート環境には、`lumount` または `mount` を使用してマウントされたパーティションを含むことはできません。

▼ ブート環境を比較する (キャラクタインタフェース)

1. メインメニューから「**Compare**」を選択します。
2. 「**Compare to Original**」または「**Compare to an Active Boot Environment**」を選択します。
3. **F3** を押します。
4. 次に示すように、元の (アクティブ) ブート環境の名前、非アクティブブート環境の名前、およびファイルのパスを入力します。

```
Name of Parent: solaris8
Name of Child: solaris8-1
Full Pathname of the file to Store Output: /tmp/compare
```

5. **F3** を押してファイルに保存します。
「**Compare**」メニューに以下の属性が表示されます。
 - 「モード」
 - 「リンクの数」
 - 「所有者」
 - 「グループ」
 - チェックサム - 指定されたブート環境内のファイルとこれに対応するアクティブブート環境内のファイルが、前述したすべてのフィールドにおいて一致する場合だけ、チェックサムを計算します。すべて一致するがチェックサムは異なるという場合には、異なるチェックサムが比較対象ファイルのエントリに付加されます。
 - 「サイズ」
 - いずれか一方のブート環境だけに存在するファイル
6. **F3** を押して「**Compare**」メニューに戻ります。

▼ ブート環境を比較する (コマンド行インタフェース)

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

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

<code>-i infile</code>	<code>infile</code> 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 <code>-t</code> のいずれか一方を使用できません (両方は使用できません)。
<code>-t</code>	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 <code>file(1)</code> 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 <code>-i</code> のいずれか一方を使用できます (両方は使用できません)。
<code>-o outfile</code>	相違についての出力を <code>outfile</code> にリダイレクトします。
<code>BE_name</code>	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 34-2 ブート環境の比較 (コマンド行インタフェース)

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

「Delete」メニューまたは `ludelete` を使用してください。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。完了状態のブート環境とは、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。ブート環境のステータスを確認する方法については、300 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。また、`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も削除できません。

▼ 非アクティブブート環境を削除する (キャラクタインタフェース)

1. メインメニューから「Delete」を選択します。
2. 削除する非アクティブブート環境の名前を入力します。

```
Name of boot environment: solaris8
```

指定された非アクティブブート環境が削除されます。

▼ 非アクティブブート環境を削除する (コマンド行インタフェース)

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

```
# 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 34-3 非アクティブブート環境の削除 (コマンド行インタフェース)

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

現在動作しているブート環境の名前を表示するには、「Current」メニューまたは `lucurr` コマンドを使用してください。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、300 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。

▼ アクティブブート環境の名前を表示する (キャラクタインタフェース)

- メインメニューから「Current」を選択します。
アクティブブート環境の名前または「No Boot Environments are defined」というメッセージが表示されます。

▼ アクティブブート環境の名前を表示する (コマンド行インタフェース)

- 次のコマンドを入力します。

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

例 34-4 アクティブブート環境の名前の表示 (コマンド行インタフェース)

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  
solaris8
```

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 Solaris リリースを別のリリースにアップグレードする場合などに便利です。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においてブート環境 solaris 7 を solaris 8 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場合は、「Rename」メニューまたは `lurename` コマンドを使用してください。

新しい名前に使用できるのは、シングルバイトの 8 ビット文字だけです。また、新しい名前の設定は、以下の規則に従う必要があります。

- 長さが 30 文字を超えない
- 英数字または ASCII 文字 (UNIX シェルで特別な意味を持つ文字を除く) で構成されている。sh(1) の「クォート」の節を参照してください。
- シングルバイトの 8 ビット文字だけ含む
- システム上に 1 つしか存在しない

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については、300 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。lmount または mount を使用してファイル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。

▼ 非アクティブブート環境の名前を変更する (キャラクタインタフェース)

1. メインメニューから「Rename」を選択します。
2. 名前を変更するブート環境を指定し、続いて新しい名前を入力します。
3. F3 を押して変更を保存します。

▼ 非アクティブブート環境の名前を変更する (コマンド行インタフェース)

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

```
# 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、この説明は長さや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。

- ブート環境を作成する場合 (-A オプション指定で `lucreate` コマンドを使用する)
- ブート環境の作成後 (`ludesc` コマンドを使用する)

`lucreate` コマンドで -A オプションを使用する方法の詳細は、265 ページの「ブート環境を初めて作成する (コマンド行インタフェース)」を参照してください。ブート環境の作成後に説明を作成する方法については、以下の説明または `ludesc (1M)` のマニュアルページを参照してください。

▼ テキストを使用してブート環境名の説明を作成または変更する方法

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

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

`-n BE_name 'BE_description'` ブート環境名と、その名前に関連付ける新しい説明を指定します。

例 34-5 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ファイルで記述されます。

```
# /usr/sbin/ludesc -n second_disk 'Solaris 9 test build, January 2001.'
```

▼ ファイルを使用してブート環境名の説明を作成または変更する方法

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

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

`-n BE_name` ブート環境名と、その名前に関連付ける新しい説明を指定します。

`-f` ブート環境名に関連付けるファイルを指定します。

`file_name`

例 34-6 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
```

▼ テキストで記述された説明からブート環境名を確認する方法

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

```
# /usr/sbin/ludesc -A 'BE_description'
```

`-A 'BE_description'` 説明に関連付けられているブート環境名を表示します。

例 34-7 説明からブート環境名を確認する

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

```
# /usr/sbin/ludesc -A 'Solaris 9 test build, January 2001.'
second_disk
```

▼ ファイル内の説明からブート環境名を確認する方法

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

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

`-f file_name` ファイル内の説明に関連付けられているブート環境名を表示します。

例 34-8 ファイル内の説明からブート環境名を確認する

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` を確認しています。

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

▼ 名前からブート環境説明を確認する方法

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

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

`-n BE_name` ブート環境名に関連付けられている説明を表示します。

例 34-9 ブート環境名から説明を確認する

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

```
# /usr/sbin/ludesc -n second_diskSolaris 9 test build, January 2001
```

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、「List」メニューまたは `lufslist` コマンドを使用してください。出力される情報は、各ブート環境マウントポイント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サイズです。

▼ 非アクティブブート環境の構成を表示する (キャラクターインタフェース)

1. メインメニューから「List」を選択します。
2. 構成を表示したいブート環境の名前を入力します。

```
Name of Boot Environment: solaris8
```

3. **F3** を押します。
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

4. **F6** を押して「List」メニューに戻ります。

▼ ブート環境の構成を表示する (コマンド行インタフェース)

1. スーパーユーザーとしてログインします。
2. 次のコマンドを入力します。

```
# lufslist
```

```
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
```


第 35 章

Solaris Live Upgrade の使用例

この章では、CUI (キャラクタユーザーインタフェース) または CLI (コマンド行インタフェース) を使用してブート環境の作成、アップグレード、およびアクティブ化を行う例を示します。また、本来のブート環境に戻す例も紹介します。

Solaris Live Upgrade によるアップグレードの使用例 (コマンド行インタフェース)

この例では、Solaris 2.6 リリースを使用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ています。この新しいブート環境は、`luupgrade` コマンドを使用して Solaris 9 にアップグレードされ、次に、`luactivate` コマンドによってアクティブ化されます。この節では、以前のブート環境にフォールバックする例も示します。

アクティブブート環境で Live Upgrade をインストールする

1. Solaris 9 DVD または Solaris 9 SOFTWARE 2 of 2 CD を挿入します。
2. 使用しているメディアに合わせて操作を行います。
 - Solaris 9 DVD を使用している場合は、`Installers` 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_9/Tool/Installers
# ./liveupgrade20
```

Solaris Web Start インストーラが表示されます。

- Solaris 9 SOFTWARE 2 of 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
```

Solaris Web Start インストーラが表示されます。

3. 「インストール形式の選択 (Select Type of Install)」パネルで「カスタム (Custom)」をクリックします。
4. 「ロケールの選択 (Locale Selection)」パネルで、インストールする言語をクリックします。
5. 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 (Component Selection)」パネルの「次へ (Next)」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 (Product Selection)」パネルの Solaris Live Upgrade の項目で「デフォルトインストール (Default Install)」をクリックします。
6. Solaris Web Start の指示に従って、ソフトウェアをインストールします。

ブート環境を作成する

-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、ブート環境を作成するためにシステム上で初めて Solaris Live Upgrade を使用する場合に必要となります。その後のブート環境作成では、-c オプションを使用する必要はありません。

新しいブート環境の名前は c0t15d0s0 です。-A オプションを用いて、このブート環境名に関連付けられ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、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 c0t4d0s0 -m /:/dev/dsk/c0t15d0s0:ufs \
-m -:/dev/dsk/c0t15d0s1:swap -n c0t15d0s0
```

非アクティブブート環境をアップグレードする

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され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/ins3-svr/export/s9/combined.s9s_wos
```

ブート環境がブート可能か確認する

lustatus コマンドは、ブート環境の作成が完了したかどうか、およびそのブート環境がブート可能であるかを報告します。

```
# lustatus

BE_name                Complete  Active  ActiveOnReboot  CopyStatus
-----
c0t4d0s0                yes      yes     yes              -
c0t15d0s0               yes      no      no               -
```

非アクティブブート環境をアクティブにする

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、c0t4d0s0 ブート環境が非アクティブになります。

```
# luactivate c0t15d0s0
# init 6
```

ソースブート環境へフォールバックする

新しいブート環境のアクティブ化の状況に応じて、次の3つの作業からいずれかを選択します。

- アクティブ化が正常に行われたが元のブート環境に戻りたいという場合は、317 ページの「ブート環境作成は正常に完了したが元のブート環境にフォールバックさせる場合」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、318 ページの「ブート環境のアクティブ化に失敗したため元のブート環境にフォールバックさせる場合」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、318 ページの「DVD、CD、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フォールバックさせる場合」を参照してください。

ブート環境作成は正常に完了したが元のブート環境にフォールバックさせる場合

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。

```
# /usr/sbin/luactivate first_disk
# init 6
```

ブート環境のアクティブ化に失敗したため元のブート環境にフォールバックさせる場合

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。

DVD、CD、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フォールバックさせる場合

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
# umount /mnt
# init 6
```

Solaris Live Upgrade によるアップグレードの例 (キャラクタインタフェース)

この例では、Solaris 2.6 リリースを使用しているシステム上に新しいブート環境を作成しています。Solaris 9 リリースにアップグレードし、アップグレードされたこのブート環境をアクティブにしています。

アクティブブート環境で Live Upgrade をインストールする

1. Solaris 9 DVD または Solaris 9 SOFTWARE 2 of 2 CD を挿入します。
2. 使用しているメディアのインストーラを実行します。

- Solaris 9 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てからインストーラを実行します。

```
# cd /cdrom/cdrom0/Solaris_9/Tool/Installers
# ./liveupgrade20
```

Solaris Web Start インストーラが表示されます。

- Solaris 9 SOFTWARE 2 of 2 CD を使用する場合は、インストーラを実行します。

```
% ./installer
```

Solaris Web Start インストーラが表示されます。

3. 「インストール形式の選択 (Select Type of Install)」パネルで「カスタム (Custom)」をクリックします。
4. 「ロケールの選択 (Locale Selection)」パネルで、インストールする言語をクリックします。
5. 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 (Component Selection)」パネルの「次へ (Next)」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 (Product Selection)」パネルの Solaris Live Upgrade の項目で「デフォルトインストール (Default Install)」を選択します。
6. Solaris Web Start の指示に従って、ソフトウェアをインストールします。

ブート環境を作成する

この例では、ソースブート環境の名前は c0t4d0s0 です。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、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。

1. キャラクタインタフェースを表示します。

```
# /usr/sbin/lu
```

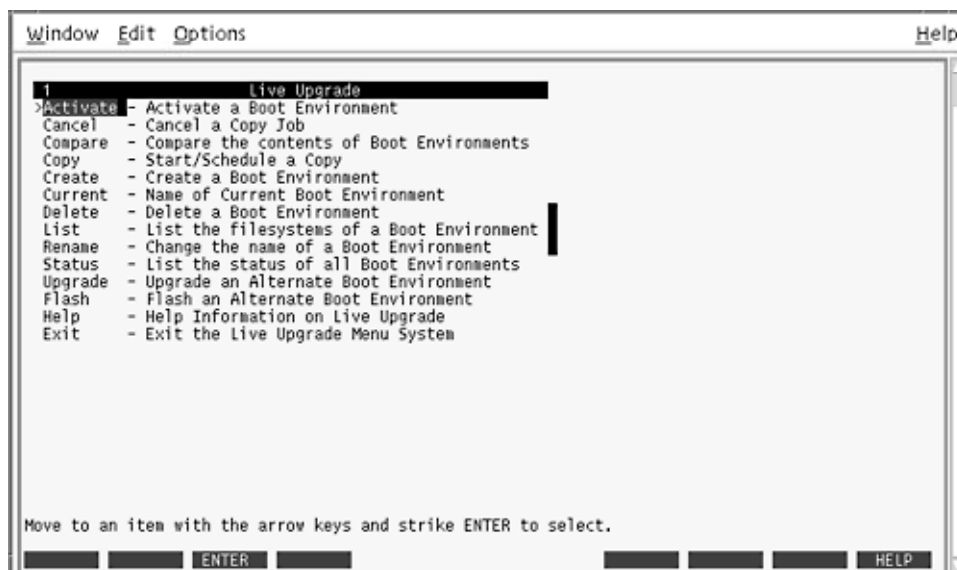


図 35-1 Solaris Live Upgrade のメインメニュー

2. メインメニューから「Create」を選択します。

Name of Current Boot Environment: **c0t4d0s0**

Name of New Boot Environment: **c0t15d0s0**

3. F3 を押します。

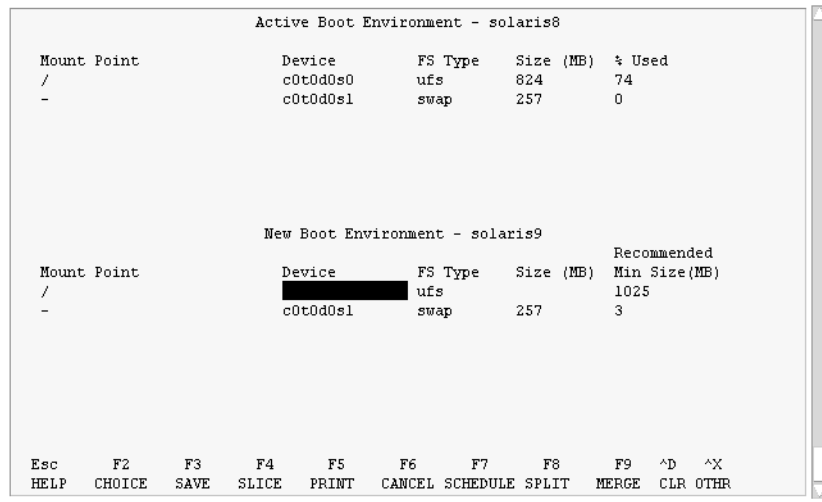


図 35-2 Solaris Live Upgrade の「Configuration」メニュー

4. 「Configuration」メニューで F2 を押して「Choices」メニューを表示します。
5. ディスク c0t15d0 からルート (/) 用としてスライス 0 を選択します。
6. 構成メニューで、分割するスワップスライスを選択して c0t15d0 上にスワップ用の新しいスライスを作成します。
7. F2 を押して「Choices」メニューを表示します。
8. 新しいスワップスライスとして、ディスク c0t15d0 からスライス 1 を選択します。
9. F3 を押して新しいブート環境を作成します。

非アクティブブート環境をアップグレードする

アップグレードのためにネットワークからオペレーティングシステムイメージを取得します。

1. メインメニューから「Upgrade」を選択します。

```
Name of New Boot Environment:  c0t15d0s0
Package Media: /net/ins3-svr/export/s9/combined.s9s_wos
```

2. F3 を押します。

非アクティブブート環境をアクティブにする

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、c0t4d0s0 ブート環境が非アクティブになります。

1. メインメニューから「Activate」を選択します。

```
Name of Boot Environment: c0t15d0s0  
Do you want to force a Live Upgrade sync operations: no
```

2. F3 を押します。
3. Return キーを押します。
4. 次のコマンドを入力します。

```
# init 6
```

フォールバックが必要な場合は、前述の例のコマンド行による作業を行います。317 ページの「ソースブート環境へフォールバックする」を参照してください。

第 36 章

Solaris Live Upgradeのコマンドに関するリファレンス情報

次の表に、メニューを使用せずにコマンド行で実行できるコマンドを示します。Solaris Live Upgradeには、次の表に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います。

表 36-1 Solaris Live Upgrade のコマンド

作業	コマンド
非アクティブブート環境をアクティブにする	luactivate (1M)
スケジュールされた処理 (コピーまたは作成) を取り消す	lucancel (1M)
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する	lucompare (1M)
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す	lucopy
ブート環境を作成する	lucreate (1M)
アクティブブート環境に名前を付ける	>lucurr (1M)
ブート環境を削除する	ludelete (1M)
ブート環境の名前に記述を追加する	ludesc (1M)
各ブート環境の重要なファイルシステムを表示する	lufslist (1M)
ブート環境に存在するすべてのファイルシステムをマウントできる。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る	lumount (1M)
ブート環境の名前を変更する	lurename (1M)
すべてのブート環境のステータスを表示する	lustatus (1M)

表 36-1 Solaris Live Upgrade のコマンド (続き)

作業	コマンド
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する。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る	luumount (1M)
非アクティブブート環境上のオペレーティング環境をアップグレードするか、あるいは非アクティブブート環境上にフラッシュアーカイブをインストールする	luupgrade (1M)

第 37 章

Solaris ソフトウェアに関するリファレンス情報

この節では、Solaris CD、ロケール、パッケージなどのリファレンス情報について説明します。

第 38 章	Solaris 9 のメディアキットに含まれる主な DVD と CD について説明します。
第 39 章	さまざまなハードウェアプラットフォームのプラットフォーム名とグループのリストを示します。
第 40 章	locale キーワードをプロファイルに設定したり、ロケールを事前設定したりするときに必要な値のリストを示します。

第 38 章

Solaris 9 メディアの構成に関するリファレンス情報

この章では、Solaris 9 ソフトウェアのメディアキットに含まれる主な DVD メディアおよび CD メディアについて説明します。

Solaris 9 メディア

次の表に、Solaris 9 リリースの主な DVD メディアおよび CD メディアのリストを示します。製品には、英語、その他の言語、およびロケールソフトウェアの DVD メディアまたは CD メディアが入ったマルチリンガル版メディアキットが入っています。

表 38-1 マルチリンガル版 DVD メディア

DVD タイトル	説明
Solaris 9 DVD	<p>次のものが入っています。</p> <ul style="list-style-type: none">■ Solaris 製品および部分ロケールをインストールするためのソフトウェア、ツール、および構成情報■ サポートおよび未サポートのソフトウェアが入っている ExtraValue ソフトウェア■ インストーラ■ 一部のサードパーティパブリックドメインソフトウェアのソースコード■ ローカライズされたインタフェースソフトウェアとドキュメント■ 英語、ヨーロッパ言語、アジア言語 (日本語を含む) 用の Solaris マニュアルセット <p>ディレクトリ構造については 図 38-1 を参照してください。</p>

表 38-2 マルチリンガル版 CD メディア

CD タイトル	説明
次に示す CD のいずれかが提供されます。 <ul style="list-style-type: none"> ■ Solaris 9 INSTALLATION CD ■ Solaris 9 MULTILINGUAL INSTALLATION CD 	<ul style="list-style-type: none"> ■ Solaris ソフトウェアをインストールするスクリプトが入っています。ディレクトリ構造については、図 38-2 を参照してください。 ■ Solaris ソフトウェアおよびすべての部分ロケールをインストールするスクリプトが入っています。ディレクトリ構造については、図 38-2 を参照してください。
Solaris 9 SOFTWARE 1 of 2 CD	Solaris 製品のインストールに必要なソフトウェア、ツール、および構成情報が入っています。ディレクトリ構造については、図 38-3 を参照してください。
Solaris 9 SOFTWARE 2 of 2 CD	<p>次のものが含まれます。</p> <ul style="list-style-type: none"> ■ 必要に応じてインストールするソフトウェア用のパッケージ。ソフトウェアごとにインストールするかどうかの指示が表示されます。 ■ サポートまたは未サポートのソフトウェアが入っている ExtraValue ソフトウェア ■ インストーラ ■ 一部のサードパーティパブリックドメインソフトウェアのソースコード <p>ディレクトリ構造については、図 38-4 を参照してください。</p> <p>必要があれば、インストールプログラムによってこの CD を求めるプロンプトが表示されます。</p>
Solaris 9 LANGUAGES CD	<p>ローカライズされたインタフェースソフトウェアとマニュアルが入っています。ディレクトリ構造については、図 38-5 を参照してください。</p> <p>特定の地域の言語をサポートするために必要な場合、インストールプログラムによってこの CD を求めるプロンプトが表示されます。</p>
Solaris 9 DOCUMENTATION 1 of 2 CD	英語およびヨーロッパ言語の Solaris マニュアルセットが入っています。
Solaris 9 DOCUMENTATION 2 of 2 CD	アジア言語 (日本語を含む) の Solaris マニュアルセットが入っています。

Solaris 9 メディアのディレクトリ構造

この節では、各 DVD と CD のディレクトリ構造について説明します。

Solaris 9 DVD のディレクトリ構造

Solaris 9 DVD のディレクトリ構造は以下のとおりです。

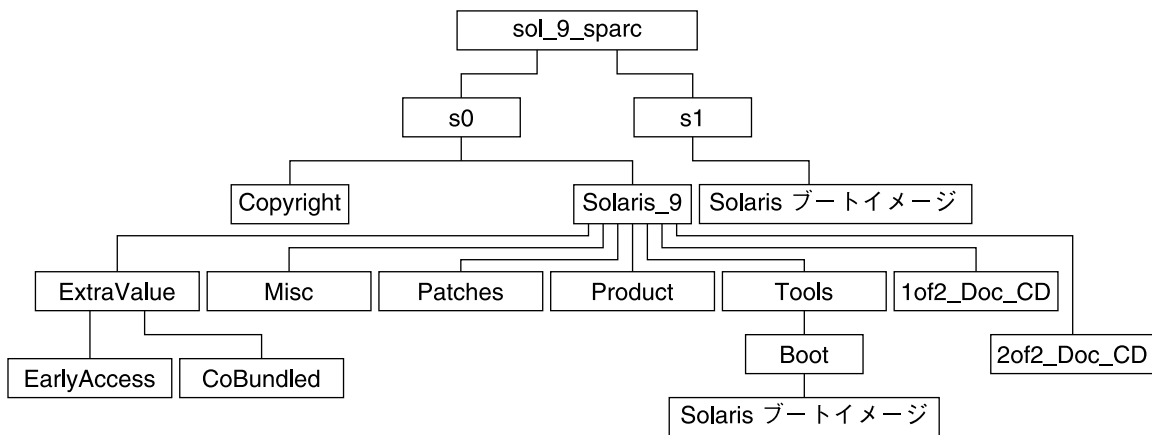


図 38-1 Solaris 9 DVD

- スライス 0 (s0) には、Copyright と Solaris_9 ディレクトリが入っています。Solaris_9 ディレクトリには、Solaris 9 ソフトウェア製品のインストールに必要なツール、ソフトウェア、および構成情報が入っています。スライス 0 には、次のディレクトリが入っています。
 - ExtraValue – 次の 2 つのサブディレクトリが入っています。
 - CoBundled – Solaris オペレーティング環境には直接的には含まれないが、サポート対象の Solaris 製品 (Solaris Web Start Wizards SDK 3.0.1 および SunScreen 3.2)
 - EarlyAccess – サポート対象外の評価用ソフトウェア
 - Misc – JumpStart のソフトウェアとファイル (rules ファイル、check スクリプト、プロファイル、begin スクリプト、finish スクリプトなど) を含む jumpstart_sample ディレクトリ
 - Patches – このリリースの時点で入手可能なすべての Solaris 9 パッチ
 - Product – Solaris 9 パッケージと制御ファイル。
 - Tools – Solaris 9 インストールツール。次のものが入っています。
 - Boot サブディレクトリ - Solaris 9 ミニルートが入っています。

- スクリプト `add_install_client`、`dial`、`rm_install_client`、および `setup_install_server`
- Installer サブディレクトリ - Solaris Live Upgrade 用のインストーラが入っています。Solaris Live Upgrade は Solaris ソフトウェアをインストールするときに自動的にインストールされます。しかし、Solaris 9 より以前のリリースから Solaris Live Upgrade を使用して Solaris 9 リリースにアップグレードするには、まず、これらのパッケージを現在の Solaris リリースにインストールする必要があります。Solaris Live Upgrade をインストールする方法については、257 ページの「Solaris Live Upgrade をインストールする方法」を参照してください。
- 1of2_Doc_CD - 英語およびヨーロッパ言語のマニュアル
- 2of2_Doc_CD - アジア言語のマニュアル
- スライス 1 (s1) には、Solaris 9 ミニルートが入っています。

Solaris 9 INSTALLATION CD のディレクトリ構造

次に、Solaris 9 MULTILINGUAL INSTALLATION CD または Solaris 9 INSTALLATION CD のディレクトリ構造を示します。

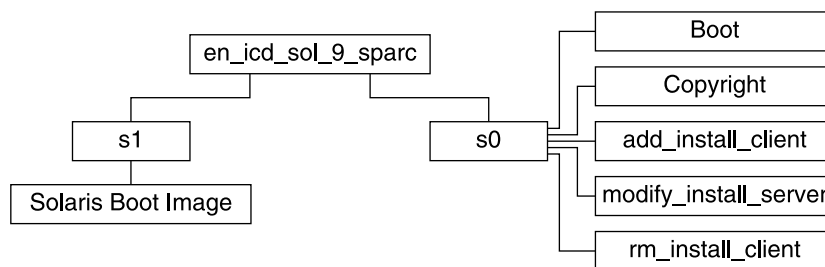


図 38-2 Solaris 9 INSTALLATION CD

- スライス 0 (s0) には、Solaris ソフトウェアをインストールするスクリプトと、Solaris ミニルートが入っている Boot サブディレクトリが入っています。次のようなスクリプトが入っています。
 - `add_install_client`
 - `modify_install_server`
 - `rm_install_client`
- スライス 1 (s1) には、Solaris 9 ミニルートが入っています。

Solaris 9 SOFTWARE CD のディレクトリ構造

次に、Solaris 9 SOFTWARE のディレクトリ構造を示します。

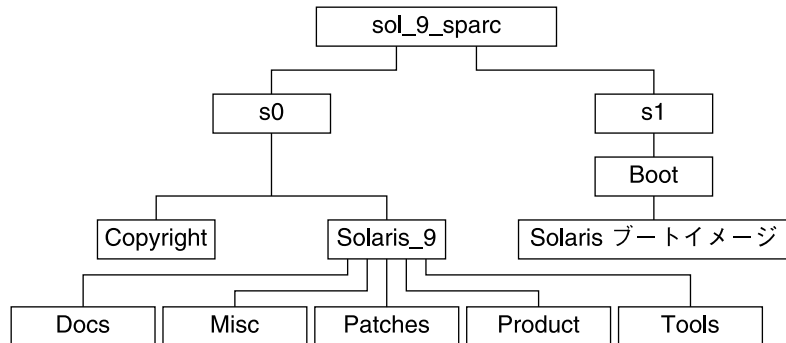


図 38-3 Solaris 9 SOFTWARE 1 of 2 CD

- スライス 0 (s0) には、Copyright と Solaris_9 ディレクトリが入っています。Solaris_9 ディレクトリには、少なくとも Solaris 9 ソフトウェア製品 (コアシステムサポートおよびエンドユーザーシステムサポート) をインストールするために必要なすべてのツール、ソフトウェア、および構成情報が入っています。スライス 0 には、次のディレクトリが入っています。
 - Docs - 空のディレクトリ
 - Misc - JumpStart のソフトウェアとファイル (rules ファイル、check スクリプト、プロファイル、begin スクリプト、finish スクリプトなど) を含む jumpstart_sample ディレクトリ
 - Patches - Solaris 9 SOFTWARE 1 of 2 CD が作成された時点で入手可能なすべての Solaris 9 パッチ
 - Product - Solaris 9 パッケージと制御ファイル
 - Tools - Solaris 9 インストールツール (add_install_client、dial、rm_install_client、setup_install_server など)。
- スライス 1 (s1) には、Solaris 9 ミニルートが入っています。

次の図に、Solaris 9 SOFTWARE 2 of 2 CD のディレクトリ構造を示します。

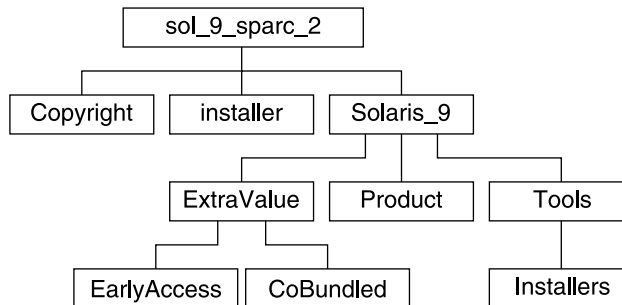


図 38-4 Solaris 9 SOFTWARE 2 of 2 CD

sol_9_sparc_2 ディレクトリには、Copyright、Solaris Web Start installer、および Solaris_9 ディレクトリが入っています。Solaris_9 ディレクトリには次のものが入っています。

- ExtraValue – 次の 2 つのサブディレクトリが入っています。
 - CoBundled – Solaris オペレーティング環境には含まれないが、サポート対象の Solaris 製品 (Solaris Web Start Wizards SDK 3.0.1 および SunScreen 3.2)
 - EarlyAccess – サポート対象外の評価用ソフトウェア
- Product – 開発者システムサポートおよび全体ディストリビューション、および全体ディストリビューションと OEM サポートのソフトウェアグループ用パッケージ
- Tools – Solaris 9 インストールツール。次のものが入っています。
 - add_to_install_server スクリプト
 - Installer サブディレクトリ - Solaris Live Upgrade 用のインストーラが入っています。Solaris Live Upgrade は Solaris ソフトウェアをインストールするときに自動的にインストールされます。しかし、Solaris 9 より前のリリースから Solaris Live Upgrade を使用して Solaris 9 リリースにアップグレードするには、まず、これらのパッケージを現在の Solaris リリースにインストールする必要があります。Solaris Live Upgrade をインストールする方法については、257 ページの「Solaris Live Upgrade をインストールする方法」を参照してください。

Solaris 9 LANGUAGES CD のディレクトリ構造

次の図に、Solaris 9 LANGUAGES CD のディレクトリ構造を示します。

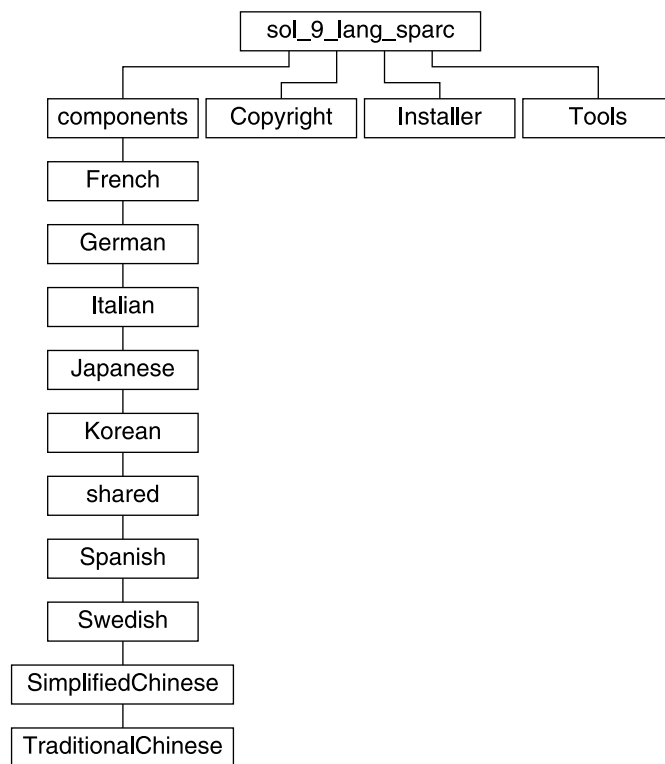


図 38-5 Solaris 9 LANGUAGES CD

sol_9_lang_sparc ディレクトリには、Solaris の言語ソフトウェアとロケールソフトウェアをインストールする Solaris Web Start インストーラが入っています。sol_9_lang_sparc ディレクトリには次のディレクトリが入っています。

- components – ロケールパッケージ (簡体字中国語、フランス語、ドイツ語、イタリア語、日本語、韓国語、スペイン語、スウェーデン語、繁体字中国語)、およびすべてのロケールで共有されるパッケージが入っています。
- Copyright – 著作権ページ
- installer – Solaris Web Start インストーラ
- Tools – インストールサーバーを作成するための add_to_install_server スクリプト

ロケールソフトウェアについての詳細は、『国際化対応言語環境の利用ガイド』を参照してください。

第 39 章

プラットフォーム名とプラットフォームグループに関するリファレンス情報

ネットワークシステムにクライアントを追加するときにはシステムアーキテクチャ (プラットフォームグループ) を、カスタム JumpStart インストール rules ファイルを作成するときにはプラットフォーム名を知っている必要があります。

プラットフォーム名とプラットフォームグループの例を下記の表に示します。全システムのリストについては、『Solaris 9 Sun ハードウェアマニュアル』を参照してください。

表 39-1 プラットフォーム名とプラットフォームグループの例

システム	プラットフォーム名	プラットフォームグループ
Ultra 5	SUNW, Ultra-5_10	sun4u
SPARCstation 20	SUNW, SPARCstation-20	sun4m

注 - システムが動作している場合、システムのプラットフォーム名は `uname -i`、システムのプラットフォームグループは `uname -m` でそれぞれ調べることができます。

第 40 章

ロケールの値 (リファレンス)

以下に、プロファイル内で `locale` キーワードを設定する場合や、ロケールを事前構成する場合に必要な値を示します。

「ロケール」によって、特定の言語と地域についてのオンライン情報を表示する方法が決定されます。1つの言語でも、日付と時間の表記、綴り、通貨など、地域によって異なるロケールが含まれます。ロケールの詳細は、『国際化対応言語環境の利用ガイド』を参照してください。

表 40-1 アジア

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
hi_IN.UTF-8	英語	インド	UTF-8 ¹	ヒンディ語 (UTF-8) Unicode 3.1
ja	日本語	日本	eucJP ²	日本語 (EUC) JIS X 0201-1976 JIS X 0208-1990 JIS X 0212-1990
ja_JP.eucJP	日本語	日本	eucJP	日本語 (EUC) JIS X 0201-1976 JIS X 0208-1990 JIS X 0212-1990
ja_JP.PCK	日本語	日本	PCK ³	日本語 (PC kanji)

¹UTF-8 は、ISO/IEC 10646-1:2000 および Unicode 3.1 で定義されている UTF-8 です。

²eucJP は、日本語 EUC コードセットを示します。ja_JP.eucJP ロケールの仕様は UI_OSF 日本語環境実装規約 Version 1.1 に準拠しており、ja ロケールは過去の Solaris リリースにおける従来の仕様に準拠しています。

³PCK は シフト JIS (SJIS) とも呼ばれます。

表 40-1 アジア (続き)

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
				JIS X 0201-1976
				JIS X 0208-1990
ja_JP.UTF-8	日本語	日本	UTF-8	日本語 (UTF-8) Unicode 3.1
ko_KR.EUC	韓国語	韓国	1001	韓国語 (EUC) KS X 1001
ko_KR.UTF-8	韓国語	韓国	UTF-8	韓国語 (UTF-8) Unicode 3.1
th_TH.UTF-8	英語	タイ	UTF-8	タイ語 (UTF-8) Unicode 3.1
th_TH.TIS620	英語	タイ	TIS620.2533	タイ語 TIS620.2533
zh_CN.EUC	簡体字中国語	中国	gb2312 ⁴	簡体字中国語 (EUC) GB2312-1980
zh_CN.GBK	簡体字中国語	中国	GBK ⁵	簡体字 (GBK)
zh_CN.GB18030	簡体字中国語	中国	GB18030-2000	簡体字中国語 (GB18030-2000) GB18030-2000
zh_CN.UTF-8	簡体字中国語	中国	UTF-8	簡体字中国語 (UTF-8) Unicode 3.1
zh_HK.BIG5HK	繁体字中国語	香港	Big5+HKSCS	繁体字中国語 (BIG5+HKSCS)
zh_HK.UTF-8	繁体字中国語	香港	UTF-8	繁体字中国語 (UTF-8) Unicode 3.1
zh_TW.EUC	繁体字中国語	台湾	cns11643	繁体字中国語 (EUC) CNS 11643-1992
zh_TW.BIG5	繁体字中国語	台湾	BIG5	繁体字中国語 (BIG5)
zh_TW.UTF-8	繁体字中国語	台湾	UTF-8	繁体字中国語 (UTF-8) Unicode 3.1

⁴gb2312 は、簡体字中国語 EUC コードセットを示します。これには GB 1988-80 および GB 2312-80 が含まれます。

⁵GBK には GB 拡張付き簡体字が含まれます。すべての GB 2312-80 文字、ISO/IEC 10646-1 のすべての Unified Han 文字、および日本語のひらがな文字、カタカナ文字を含みます。また、中国語、日本語、韓国語の文字セット、および ISO/IEC 10646-1 の多くの文字も含みます。

表 40-2 オーストラレーシア

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
en_AU.ISO8859-1	英語	オーストラリア	ISO8859-1	英語 (オーストラリア)
en_NZ.ISO8859-1	英語	ニュージーランド	ISO8859-1	英語 (ニュージーランド)

表 40-3 中央アメリカ

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
es_CR.ISO8859-1	スペイン語	コスタリカ	ISO8859-1	スペイン語 (コスタリカ)
es_GT.ISO8859-1	スペイン語	グアテマラ	ISO8859-1	スペイン語 (グアテマラ)
es_NI.ISO8859-1	スペイン語	ニカラグア	ISO8859-1	スペイン語 (ニカラグア)
es_PA.ISO8859-1	スペイン語	パナマ	ISO8859-1	スペイン語 (パナマ)
es_SV.ISO8859-1	スペイン語	エルサルバドル	ISO8859-1	スペイン語 (エルサルバドル)

表 40-4 中央ヨーロッパ

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
cs_CZ.ISO8859-2	英語	チェコ共和国	ISO8859-2	チェコ語 (チェコ)
de_AT.ISO8859-1	ドイツ語	オーストリア	ISO8859-1	ドイツ語 (オーストリア)
de_AT.ISO8859-15	ドイツ語	オーストリア	ISO8859-15	ドイツ語 (オーストリア、ISO8859-15 - Euro)
de_CH.ISO8859-1	ドイツ語	スイス	ISO8859-1	ドイツ語 (スイス)
de_DE.UTF-8	ドイツ語	ドイツ	UTF-8	ドイツ語 (ドイツ、Unicode 3.1)
de_DE.ISO8859-1	ドイツ語	ドイツ	ISO8859-1	ドイツ語 (ドイツ)
de_DE.ISO8859-15	ドイツ語	ドイツ	ISO8859-15	ドイツ語 (ドイツ、ISO8859-15 - Euro)
fr_CH.ISO8859-1	フランス語	スイス	ISO8859-1	フランス語 (スイス)
hu_HU.ISO8859-2	英語	ハンガリー	ISO8859-2	ハンガリー語 (ハンガリー)
pl_PL.ISO8859-2	英語	ポーランド	ISO8859-2	ポーランド語 (ポーランド)
pl_PL.UTF-8	英語	ポーランド	UTF-8	ポーランド語 (ポーランド、Unicode 3.1)
sk_SK.ISO8859-2	英語	スロバキア	ISO8859-2	スロバキア語 (スロバキア)

表 40-5 東ヨーロッパ

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
bg_BG.ISO8859-5	英語	ブルガリア	ISO8859-5	ブルガリア語 (ブルガリア)

表 40-5 東ヨーロッパ (続き)

ロケール	ユーザーインターフェース	地域	コードセット	言語サポート
et_EE.ISO8859-15	英語	エストニア	ISO8859-15	エストニア語 (エストニア)
hr_HR.ISO8859-2	英語	クロアチア	ISO8859-2	クロアチア語 (クロアチア)
lt_LT.ISO8859-13	英語	リトアニア	ISO8859-13	リトアニア語 (リトアニア)
lv_LV.ISO8859-13	英語	ラトビア	ISO8859-13	ラトビア語 (ラトビア)
mk_MK.ISO8859-5	英語	マケドニア	ISO8859-5	マケドニア語 (マケドニア)
ro_RO.ISO8859-2	英語	ルーマニア	ISO8859-2	ルーマニア語 (ルーマニア)
ru_RU.KOI8-R	英語	ロシア	KOI8-R	ロシア語 (ロシア、KOI8-R)
ru_RU.ANSI1251	英語	ロシア	ansi-1251	ロシア語 (ロシア、ANSI 1251)
ru_RU.ISO8859-5	英語	ロシア	ISO8859-5	ロシア語 (ロシア)
ru_RU.UTF-8	英語	ロシア	UTF-8	ロシア語 (ロシア、Unicode 3.1)
sh_BA.ISO8859-2@bosnia	英語	ボスニア	ISO8859-2	ボスニア語 (ボスニア)
sl_SI.ISO8859-2	英語	スロベニア	ISO8859-2	スロベニア語 (スロベニア)
sq_AL.ISO8859-2	英語	アルバニア	ISO8859-2	アルバニア語 (アルバニア)
sr_YU.ISO8859-5	英語	セルビア	ISO8859-5	セルビア語 (セルビア)
tr_TR.ISO8859-9	英語	トルコ	ISO8859-9	トルコ語 (トルコ)
tr_TR.UTF-8	英語	トルコ	UTF-8	トルコ語 (トルコ、Unicode 3.1)

表 40-6 中東

ロケール	ユーザーインターフェース	地域	コードセット	言語サポート
He	英語	イスラエル	ISO8859-8	ヘブライ語 (イスラエル)

表 40-7 北アフリカ

ロケール	ユーザーインタ フェース	地域	コードセット	言語サポート
ar_EG.UTF-8	英語	エジプト	UTF-8	アラビア語 (エジプト)
Ar	英語	エジプト	ISO8859-6	アラビア語 (エジプト)

表 40-8 北アメリカ

ロケール	ユーザーインタ フェース	地域	コードセット	言語サポート
en_CA.ISO8859-1	英語	カナダ	ISO8859-1	英語 (カナダ)
en_US.ISO8859-1	英語	米国	ISO8859-1	英語 (U.S.A.)
en_US.ISO8859-15	英語	米国	ISO8859-15	英語 (U.S.A.、ISO8859-15 - Euro)
en_US.UTF-8	英語	米国	UTF-8	英語 (U.S.A.、Unicode 3.1)
fr_CA.ISO8859-1	フランス語	カナダ	ISO8859-1	フランス語 (カナダ)
es_MX.ISO8859-1	スペイン語	メキシコ	ISO8859-1	スペイン語 (メキシコ)

表 40-9 北ヨーロッパ

ロケール	ユーザーインタ フェース	地域	コードセット	言語サポート
da_DK.ISO8859-1	英語	デンマーク	ISO8859-1	デンマーク語 (デンマーク)
da_DK.ISO8859-15	英語	デンマーク	ISO8859-15	デンマーク語 (デンマーク、ISO8859-15-Euro)
fi_FI.ISO8859-1	英語	フィンランド	ISO8859-1	フィンランド語、Unicode 3.1)
fi_FI.ISO8859-15	英語	フィンランド	ISO8859-15	フィンランド語 (フィンランド、ISO8859-15-Euro)
fi_FI.UTF-8	英語	フィンランド	UTF-8	フィンランド語 (フィンランド)
is_IS.ISO8859-1	英語	アイスランド	ISO8859-1	アイスランド語 (アイスランド)
no_NO.ISO8859-1@bokmal	英語	ノルウェー	ISO8859-1	ノルウェー語 (ノルウェー - ブークモール語)
no_NO.ISO8859-1@nyorsk	英語	ノルウェー	ISO8859-1	ノルウェー語 (ノルウェー - ニーノシク語)

表 40-9 北ヨーロッパ (続き)

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
sv_SE.ISO8859-1	スウェーデン語	スウェーデン	ISO8859-1	スウェーデン語 (スウェーデン)
sv_SE.ISO8859-15	スウェーデン語	スウェーデン	ISO8859-15	スウェーデン語 (スウェーデン、ISO8859-15-Euro)
sv_SE.UTF-8	スウェーデン語	スウェーデン	UTF-8	スウェーデン語 (スウェーデン、Unicode 3.1)

表 40-10 南アメリカ

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
es_AR.ISO8859-1	スペイン語	アルゼンチン	ISO8859-1	スペイン語 (アルゼンチン)
es_BO.ISO8859-1	スペイン語	ボリビア	ISO8859-1	スペイン語 (ボリビア)
es_CL.ISO8859-1	スペイン語	チリ	ISO8859-1	スペイン語 (チリ)
es_CO.ISO8859-1	スペイン語	コロンビア	ISO8859-1	スペイン語 (コロンビア)
es_EC.ISO8859-1	スペイン語	エクアドル	ISO8859-1	スペイン語 (エクアドル)
es_PE.ISO8859-1	スペイン語	ペルー	ISO8859-1	スペイン語 (ペルー)
es_PY.ISO8859-1	スペイン語	パラグアイ	ISO8859-1	スペイン語 (パラグアイ)
es_UY.ISO8859-1	スペイン語	ウルグアイ	ISO8859-1	スペイン語 (ウルグアイ)
es_VE.ISO8859-1	スペイン語	ベネズエラ	ISO8859-1	スペイン語 (ベネズエラ)
pt_BR.ISO8859-1	英語	ブラジル	ISO8859-1	ポルトガル語 (ブラジル)
pt_BR.UTF-8	英語	ブラジル	UTF-8	ポルトガル語 (ブラジル、Unicode 3.1)

表 40-11 南ヨーロッパ

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
ca_ES.ISO8859-1	英語	スペイン	ISO8859-1	カタロニア語 (スペイン)
ca_ES.ISO8859-15	英語	スペイン	ISO8859-15	カタロニア語 (スペイン、ISO8859-15 - Euro)
el_GR.ISO8859-7	英語	ギリシャ語	ISO8859-7	ギリシャ語 (ギリシャ)
es_ES.ISO8859-1	スペイン語	スペイン	ISO8859-1	スペイン語 (スペイン)

表 40-11 南ヨーロッパ (続き)

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
es_ES.ISO8859-15	スペイン語	スペイン	ISO8859-15	スペイン語 (スペイン、ISO8859-15 - Euro)
es_ES.UTF-8	スペイン語	スペイン	UTF-8	スペイン語 (スペイン、Unicode 3.1)
it_IT.ISO8859-1	イタリア語	イタリア	ISO8859-1	イタリア語 (イタリア)
it_IT.ISO8859-15	イタリア語	イタリア	ISO8859-15	イタリア語 (イタリア、ISO8859-15 - Euro)
it_IT.UTF-8	イタリア語	イタリア	UTF-8	イタリア語 (イタリア、Unicode 3.1)
pt_PT.ISO8859-1	英語	ポルトガル	ISO8859-1	ポルトガル語 (ポルトガル)
pt_PT.ISO8859-15	英語	ポルトガル	ISO8859-15	ポルトガル語 (ポルトガル、ISO8859-15 - Euro)

表 40-12 西ヨーロッパ

ロケール	ユーザーインタフェース	地域	コードセット	言語サポート
en_GB.ISO8859-1	英語	イギリス	ISO8859-1	英語 (イギリス)
en_IE.ISO8859-1	英語	アイルランド	ISO8859-1	英語 (アイルランド)
fr_BE.ISO8859-1	フランス語	ベルギーのワロン地域	ISO8859-1	フランス語 (ベルギーのワロン地域、Unicode 3.1)
fr_BE.UTF-8	フランス語	ベルギーのワロン地域	UTF-8	フランス語 (ベルギーのワロン地域、Unicode 3.1)
fr_FR.ISO8859-1	フランス語	フランス	ISO8859-1	フランス語 (フランス)
fr_FR.UTF-8	フランス語	フランス	UTF-8	フランス語 (フランス、Unicode 3.1)
nl_BE.ISO8859-1	英語	ベルギーのフラマン地域	ISO8859-1	ドイツ語 (ベルギーのフラマン地域)
nl_NL.ISO8859-1	英語	オランダ	ISO8859-1	オランダ語 (オランダ)

問題発生時の解決方法

この章では、Solaris 9 ソフトウェア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、それぞれの問題の解決方法を示します。まず、次のリストを使用して、インストールプロセスのどこで問題が発生したか確認してください。

- 345 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」
- 346 ページの「システムのブートに関する問題」
- 350 ページの「Solaris 9 オペレーティング環境の初期インストール時の問題」
- 351 ページの「Solaris 9 オペレーティング環境のアップグレード時の問題」

注 - この付録で「ブート可能なメディア」と記載されている場合、これはインストールプログラム (Solaris suninstall プログラム、Solaris Web Start プログラム、またはカスタム JumpStart) の 1 つを意味します。

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なければならない。

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。

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ない。

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 が正しく接続されているかどうか確認してください。

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らない。

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 Solaris 9 をインストールするように正しく設定されているか確認します。たとえば、設定時のシステムのプラットフォームグループの指定が正しいか確認します。

あるいは、Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2、Solaris 9 SOFTWARE 2 of 2 CD、および Solaris 9 LANGUAGES CD のイメージをインストールサーバーにコピーしなかった場合は、インストールサーバー上の Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD が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 /kernel/unix

原因: PROM の boot- file の値を明示的に /kernel/unix に指定したときに発生するエラー。

対処方法:

- PROM の boot- file の値を “ ” (無指定) に設定変更します。
- diag-switch が off と true に設定されているか確認します。

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。
- Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 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!

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。

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: 初めてカスタム 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。システムについての情報およびシステムがどのようにブートするかを調べるには、`-v` オプションを指定してブートコマンドを実行してください。`-v` オプションを使用すると、ブートコマンドは画面に詳しいデバッグ情報を表示します。

注 - このフラグを指定しなくてもメッセージは出力されますが、システムのログファイルが出力先となります。詳細は、`syslogd(1M)` を参照してください。

対処方法: `ok` プロンプトで、`ok boot net -v - install` と入力します。

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

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 Solaris 9 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、Solaris 9 Supplement CD に入っています。

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。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する。次の原因が考えられる。

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。

対処方法: ネットワーク上の複数のサーバーの /etc/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 /etc/bootparams ファイルから登録を削除します。

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。

対処方法: インストールクライアントが複数のサーバーの /tftpboot または /rplboot ディレクトリに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のエントリがあり、別のサーバーの /etc/bootparams ファイルに、以下に示すように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している。

```
* install_config=profile_server:pathpath
```

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する。

対処方法: ネームサービスの bootparams マップまたはテーブルに * install_config= などのワイルドカードを使った登録がある場合は、その登録を削除し、削除したのと同じ行をブートサーバーの /etc/bootparams ファイルに追加します。

No network boot server. Unable to install the system. See installation instructions.

原因: このエラーは、ネットワークからインストールしようとしているシステムで発生する。このシステムは、適切に設定されていない。

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。88 ページの「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

原因: このエラーはネットワークから Solaris 9 をインストールしようとしてブートソフトウェアが次のものを見つけれられない場合に発生します。

- Solaris 9 DVD またはインストールサーバー上の Solaris 9 DVD イメージコピー
- Solaris 9 SOFTWARE 1 of 2 CD またはインストールサーバー上の Solaris 9 SOFTWARE 1 of 2 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 Solaris 9 をインストールする場合は、Solaris 9 DVD または Solaris 9 SOFTWARE 1 of 2 CD が CD-ROM ドライブに挿入されてマウントされていること、および `/etc/dfs/dfstab` ファイルで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ディスク上にある Solaris 9 DVD イメージまたは Solaris 9 SOFTWARE 1 of 2 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...

原因: 1 クライアントはネットワークを介してブートしようとしているが、認識してくれるシステムを見つけることができない。

対処方法: 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 Solaris `suninstall` 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、JumpStart プログラムまたは Solaris `suninstall` プログラム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 アドレスが不正である。

対処方法: インストールサーバーの `/etc/ethers` ファイルに登録されているクライアントの Ethernet アドレスが正しいか調べます。

原因: 3 カスタム JumpStart インストールでは、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ようプラットフォームグループを `add_install_client` コマンドによって指定します。`add_install_client` を使用する際に不正な構成値を使用すると、この問題が発生します。たとえば、インストールしたマシンが `sun4u` であるのに誤って `sun4m` と指定した場合などが考えられます。

対処方法: 正しい構成値を使用して `add_install_client` を実行し直します。

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する。

対処方法: ネームサーバー上で、インストール中のシステムに対する /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。

インストールサーバーを設定して、ネットワークを介して Solaris 9 をインストールするようにシステムを構成しても、システムをブートできない。

原因: tftpd がインストールサーバー上で動作していない可能性があります。

対処方法: tftpd デーモンがインストールサーバーで実行されていることを確認します。次のコマンドを入力します。

```
# ps -ef | grep tftpd
```

このコマンドを入力しても tftpd デーモンが実行されていることを示す行が表示されない場合は、/etc/inetd.conf ファイルを編集して、次に示す行からコメント文字 (#) を削除します。

```
# tftp dgram udp wait root /usr/sbin/in.tftpd in.tftpd \  
-s /tftpboot
```

この変更を行なった後、システムをリブートしてください。

Solaris 9 オペレーティング環境の初期インストール時の問題

初期インストールが失敗する

対処方法: 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、Solaris 9 DVD、Solaris 9 INSTALLATION CD、Solaris 9 SOFTWARE 1 of 2 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。

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、このソフトウェアのインストールを解除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、Solaris インストールの処理をもう一度行う必要があります。

```
/cdrom/Solaris_9/SUNW xxxx/reloc.cpio: Broken pipe
```

説明: このエラーメッセージはインストールには影響しない。

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。

Solaris 9 オペレーティング環境のアップグレード時の問題

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に問題がある。

対処方法: /etc/vfstab ファイルの以下の行をコメントにしてください。

- 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/bzczt not found

原因: パッチクラスタが必要なために Solaris Live Upgrade が失敗している。

対処方法: Solaris Live Upgrade のインストールでパッチを使用する必要が生じました。 <http://sunsolve.sun.com> でパッチを入手してください。

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。原因としては、以下が考えられる。

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ある。

対処方法: /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 ファイルが存在しない。

対処方法: 以下の形式で 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 Solaris ソフトウェアのバージョン

原因: 3 /var/sadm/softinfo に SUNWusr が存在しない。

対処方法: 初期インストールを行う必要があります。この 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しました。

対処方法:

- ミラーでなければ、vsftab ファイル内でコメントアウトします。
- ミラーの場合は、ミラーを破棄してインストールし直します。

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: アップグレード中、インストールプログラムは、システムの /etc/vfstab ファイルにリストされているすべてのファイルシステムを、アップグレードしようとしているルートファイルシステムにマウントしようとする。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、失敗して終了する。

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、あるい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は、すべてコメントにします。Solaris suninstall プログラムはアップグレード中、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。

アップグレードが失敗する

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ない。

原因: 第 5 章を参照してディスク容量に問題がないかを確認し、自動配置機能による領域の再配置を行わずに解決できるかどうかを調べます。

ミラー化されたルートのアップグレードに関連する問題

対処方法: Solaris ボリュームマネージャでミラー化されているルートを使用してアップグレードする際に問題が発生する場合は、『Solaris ボリュームマネージャの管理』の「Solaris ボリュームマネージャの障害追跡」を参照してください。

▼ 問題発生後にアップグレードを継続する方法

アップグレードに失敗し、システムをブートできない場合があります。このような状況は、電源の故障やネットワーク接続の障害などが発生した場合に起こる可能性があります。制御できない場合に発生します。

1. **Solaris 9 DVD、Solaris 9 INSTALLATION CD、Solaris 9 SOFTWARE 1 of 2 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。

Solaris `suninstall` 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判断し、アップグレードを継続します。

Veritas VxVm の実行中に Solaris 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

実行中の Veritas VxVM で Solaris Live Upgrade を用いてアップグレードを行う場合、以下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 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. 非アクティブブート環境を作成します。259 ページの「新しいブート環境の作成」を参照してください。
2. 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する必要があります。

- 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 /.alt.12345
```

- b. 次の例のように、`vfstab` 上に存在するディレクトリに移動します。

```
# cd /.alt.12345/etc
```

- c. 次の例のように、非アクティブブート環境の `vfstab` ファイルをコピーします。

```
# cp vfstab vfstab.501
```

- d. 次の例のように、`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。

```
# sed < '/vx\/dsk\/s\/^\/#\/g' < vfstab.novxfs > vfstab
```

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. 次の例のように、変更した `vfstab` ファイルをコピーします。

```
# cp vfstab.novxfs vfstab
```

- f. 次の例のように、非アクティブブート環境のシステムファイルがあるディレクトリに移動します。

```
# cd /.alt.12345/etc
```

g. 次の例のように、非アクティブブート環境のシステムファイルをコピーします。

```
# cp system system.501
```

h. `drv/vx` を含むすべての **forceload**: エントリをコメントアウトします。

```
# 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. 次の例のように、非アクティブブート環境上の `install-db` ファイルがあるディレクトリに移動します。

```
# cd /.alt.12345/etc
```

j. 次の例のように、**Veritas** `install-db` ファイルを作成します。

```
# touch vx/reconfig.d/state.d/install-db
```

k. 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 mount_point
```

3. 非アクティブブート環境をアップグレードします。第 33 章を参照してください。

4. 非アクティブブート環境をアクティブにします。291 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。

5. システムをシャットダウンします。

```
# init 0
```

6. 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、これらは無視して構いません。非アクティブブート環境がアクティブになります。

7. **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 VRTSvman VRTSvmdev
```

8. 元の `vfstab` とシステムファイルを復元します。

```
# cp /etc/vfstab.original /etc/vfstab
# cp /etc/system.original /etc/system
```

9. リブートします。

```
# init 6
```


付録 B

リモートからのインストールまたはアップグレード

この付録では、DVD-ROM ドライブまたは CD-ROM ドライブが直接接続されていないシステムまたはドメインに、Solaris Web Start を使って Solaris オペレーティング環境をインストールまたはアップグレードする方法について説明します。

注 - マルチドメインサーバーに Solaris オペレーティング環境をインストール、またはアップグレードする場合は、インストールプロセスを開始する前に、システムコントローラまたはシステムサービスプロセッサのマニュアルを参照してください。

Solaris Web Start を使用してリモート DVD-ROM または CD-ROM からインストールまたはアップグレードする

DVD-ROM または CD-ROM ドライブが直接接続されていないシステムまたはドメインサーバーに Solaris オペレーティング環境をインストールしたい場合は、ほかのシステムに接続されたドライブを使用することができます。この場合、両方のシステムが同じサブネット上に接続されている必要があります。インストール手順は次のとおりです。

▼ リモート DVD-ROM または CD-ROM を使用してインストールまたはアップグレードを行う方法

注 - この手順では、システムでボリュームマネージャ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を使用せずにメディアを管理する方法については、Solaris 9 Beta System Administrator Collection に含まれている『Solaris のシステム管理 (基本編)』を参照してください

この手順では、DVD-ROM または CD-ROM が接続されているリモートシステムをリモートシステムと呼びます。そして、インストールするクライアントをクライアントシステムと呼びます。

1. **Solaris** オペレーティング環境が動作し、かつ **DVD-ROM** ドライブまたは **CD-ROM** ドライブを備えたシステムを特定します。
2. **DVD-ROM** ドライブまたは **CD-ROM** ドライブが存在するリモートシステムで、ドライブに **Solaris 9 DVD** または **Solaris 9 INSTALLATION CD** を挿入します。ボリュームマネージャによってディスクがマウントされます。
3. リモートシステムで、`add_install_client` コマンドが含まれている **DVD** または **CD** のディレクトリに移ります。

- DVD メディアの場合は、次のように入力します。

```
remote system# cd /cdrom/cdrom0/s0/Solaris_9/Tools
```

- CD メディアの場合は、次のように入力します。

```
remote system# cd /cdrom/cdrom0/s0
```

4. リモートシステムで、クライアントとしてインストールしたいシステムを追加します。

- DVD メディアの場合は、次のように入力します。

```
remote system# ./add_install_client \
client_system_name arch
```

- CD メディアの場合は、次のように入力します。

```
remote system# ./add_install_client -s remote_system_name: \
/cdrom/cdrom0/s0 client_system_name arch
```

`remote_system_name` DVD-ROM ドライブまたは CD-ROM ドライブを備えたシステムの名前

`client_system_name` インストールしたいマシンの名前

arch インストールしたいマシンのプラットフォームグループ (例: sun4u)
インストールしたいシステム上で、`uname -m` コマンドを実行するとプラットフォームグループが表示されます。

5. インストールしたいクライアントシステムをブートします。

```
client system: ok boot net
```

Solaris Web Start インストールが始まります。

6. 必要なら、説明に従ってシステム構成情報を入力します。

- DVD メディアを使用している場合は、画面の説明に従ってインストールを完了します。ここで作業は終了です。
- CD メディアを使用している場合は、マシンがリブートされ、Solaris インストールプログラムが始まります。「ようこそ (Welcome)」画面が表示された後、「ネットワーク接続性」画面で「ネットワークに接続 (Network File System)」を選択します。設定を続けると、「媒体の指定 (Specify Media)」画面が表示されます。手順 7 へ進みます。

7. 「媒体の指定 (Specify Media)」画面で、「次へ (Next)」をクリックします。

「ネットワークファイルシステムのパスの指定 (Specify Network Filesystem Path)」画面が表示され、テキストフィールドにインストールパスが示されます。

```
client_system_ip_address:/cdrom/cdrom0/s0
```

8. DVD または CD がマウントされているリモートシステムで、ディレクトリを / (root) に変更します。

```
remote system# cd /
```

9. リモートシステムで、共有されているスライスへのパスがあるかチェックします。

```
remote system# share
```

10. リモートシステムで、手順 9 で見つかったパスを使って Solaris 9 DVD または Solaris 9 INSTALLATION CD の共有を解除します。パスが 2 つのスライスを指している場合は、両方のスライスを unshare します。

```
remote system# unshare absolute_path
```

absolute_path share コマンドで表示される絶対パスです。

この例では、スライス 0 と 1 の共有が解除されます。

```
remote system# unshare /cdrom/cdrom0/s0
```

```
remote system# unshare /cdrom/cdrom0/s1
```

11. Solaris 9 INSTALLATION CD を取り出します。

```
remote system# eject cdrom
```

12. リモートシステムの CD-ROM ドライブに **Solaris 9 SOFTWARE 1 of 2 CD** を挿入します。

13. リモートシステムで、**Solaris 9 SOFTWARE 1 of 2 CD** をエクスポートします。

```
remote system# share -F nfs -o ro,anon=0 /cdrom/cdrom0/s0
```

14. インストールするクライアントシステムで、「次へ (Next)」をクリックして **Solaris** のインストールを継続します。

15. **Solaris Web Start** が **Solaris 9 SOFTWARE 2 of 2 CD** を挿入するようプロンプトを表示したら、手順 9 から手順 14 までを行い、**Solaris 9 SOFTWARE 1 of 2 CD** の共有解除と **Solaris 9 SOFTWARE 2 of 2 CD** のエクスポートとインストールを行います。

16. **Solaris Web Start** が **Solaris 9 LANGUAGES CD** を挿入するようプロンプトを表示したら、手順 9 から手順 14 までを行い、**Solaris 9 SOFTWARE 2 of 2 CD** の共有解除と **Solaris 9 LANGUAGES CD** のエクスポートとインストールを行います。

Solaris 9 LANGUAGES CD をエクスポートすると、CD-ROM がマウントされているシステム上にインストール画面が表示されます。**Solaris 9 LANGUAGES CD** をインストールする間は、このインストール画面は無視してください。**Solaris 9 LANGUAGES CD** のインストールが終了した後で、このインストール画面を閉じてください。

その他の SvR4 パッケージ要件に関するリファレンス情報

この付録は、カスタム JumpStart プログラムや Solaris Live Upgrade を使ってパッケージ (特にサードパーティのパッケージ) のインストールや削除を行うシステム管理者のためのものです。パッケージが SvR4 パッケージ要件に準拠していれば、カスタム JumpStart インストールを非対話的に行うことができます。さらに、動作中の現在のシステムは変更されないため、Solaris Live Upgrade を使ってアップグレードを行うことができます。

注 - 代替ルート (/) はオペレーティング環境のコピーであり、動作中のシステムではありません。

パッケージ化要件の概要

カスタム JumpStart プログラムや Solaris Live Upgrade が正しく動作するためには、パッケージが SvR4 パッケージ要件に準拠していなければなりません。『*Application Packaging Developer's Guide*』には、パッケージ化要件や用語の定義が記載されています。その中でも特に、「Advanced Package Creation Techniques」を参照してください。

パッケージやユーザー管理ファイルの追加や削除に関する基本的な情報については、『Solaris のシステム管理 (基本編)』の「ソフトウェアの管理 (概要)」を参照してください。さらに、該当するマニュアルページも参照してください。

この付録で参照するコマンドの詳細については、マニュアルページ `dircmp (1)`、`fssnap (1M)`、`ps (1)`、または `truss (1)` を参照してください。

表 C-1 に、Solaris Live Upgrade とカスタム JumpStart プログラムに適用される情報の参照先を示します。

表 C-1 要件情報

インストール方法	要件に関する参照先
Solaris Live Upgrade	<ul style="list-style-type: none"> ■ 『Application Packaging Developer's Guide』の SVR4 パッケージ要件 ■ 362 ページの「カスタム JumpStart プログラムと Solaris Live Upgrade の代替ルート (/) 要件」 ■ 364 ページの「Solaris Live Upgrade 代替ブート環境の要件」
Custom JumpStart プログラム	<ul style="list-style-type: none"> ■ 『Application Packaging Developer's Guide』の SVR4 パッケージ要件 ■ 362 ページの「カスタム JumpStart プログラムと Solaris Live Upgrade の代替ルート (/) 要件」 ■ 366 ページの「カスタム JumpStart プログラムのアップグレード要件」

カスタム JumpStart プログラムと Solaris Live Upgrade の代替ルート (/) 要件

代替ルート (/) はオペレーティング環境のコピーであり、動作中のシステムではありません。Live Upgrade やカスタム JumpStart プログラムで使用するパッケージは、次の要件に準拠していなければなりません。

- ユーザー入力なしでカスタム JumpStart インストールまたはアップグレードが可能である。
- 動作中の現在のシステムが変更されない。これは、Solaris Live Upgrade を使用する場合に必要です。

以下に、代替ルート (/) 要件に準拠する必要性を説明します。

- オペレーティングシステムのインストールを正しく行うためには、パッケージが代替ルート (/) の指定子を正しく認識する必要があります。

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せん。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) パスが両方とも指定されているパッケージは、代替ルート (/) 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 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なりません。

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、パッケージに添付されているスクリプトで、動作中のシステムを変更してはなりません。インストールスクリプトを作成する場合には、そのスクリプトで、`$PKG_INSTALL_ROOT` 変数が前に付加されたディレクトリまたはファイルを参照する必要があります。パッケージでは、書き込むすべてのディレクトリやファイルの前に `$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。表 C-2 に、スクリプトの正しい構文を示します。

表 C-2 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 の相違点の概要

`$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します。

```
# pkgadd -R /a SUNWvxxm
```

この場合、パッケージのインストール時に `$PKG_INSTALL_ROOT` の値は `/a` にセットされます。

`$BASEDIR` は、再配置可能なパッケージオブジェクトがインストールされる再配置可能なベースディレクトリを指しています。ここにインストールされるのは、再配置可能オブジェクトだけです。再配置可能でないオブジェクト (`pkgmap` ファイルに絶対パスが指定されているオブジェクト) は、代替ルート (/) の相対パスにインストールされます。`$BASEDIR` との相対パスにはインストールされません。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 (非再配置可能) 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します。

```
# pkgadd -R /a SUNWtest
```

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。

Solaris Live Upgrade 代替ブート環境の要件

Solaris Live Upgrade を使用したり、新しいブート環境を作成する場合には、次の項目に従って問題の発生を防止してください。

- パッケージに添付されているスクリプトは、アクティブなオペレーティング環境から独立していなければなりません。パッケージに添付されているスクリプトには、パッケージのインストールや削除を行う際に特定の時点で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, postremove)。これらのスクリプトは、現在のアクティブなオペレーティング環境から独立していなければなりません。これは、代替ブート環境が Solaris Live Upgrade によって切り替えられることがあり得るからです。
- これらの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てはなりません。ps や truss はオペレーティングシステムに依存し、動作中のシステムに関する情報を報告します。
- これらのスクリプトでは、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。ただし、現在の代替ルート (/) を変更する場合には、362 ページの「カスタム JumpStart プログラムと Solaris Live Upgrade の代替ルート (/) 要件」に述べる規則に従ってください。
- すべてのスクリプトは 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。
- パッケージに添付されているスクリプトでは、2.6 より前のリリースに存在していないコマンドを呼び出すことはできません。たとえば、スクリプトで pgrep コマンドを呼び出すことはできません。2.6 以降、多くのコマンドに新しい機能が追加されました。スクリプトでは、2.6 リリースにないコマンドオプション

を使用することはできません。たとえば、`-f` オプションは `umount` コマンドに新たに追加されたものです。

- 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。

たとえば、パッケージが `/export/SUNWvxxvm` にあるなら、コマンドを次のように指定します。

```
# pkgchk -d /export SUNWvxxvm
```

エラーが表示されないことを確認します。

パッケージを作成したら、`pkgadd` コマンドに `-R dir_name` オプションを指定してこれを代替ルート (`/`) にインストールしてテストします。さらに、この処理が終わったら、次のように、`pkgchk` コマンドでパッケージが正しいかチェックします。

```
# pkgadd -d . -R /a SUNWvxxvm
# pkgchk -R /a SUNWvxxvm
```

エラーが表示されないことを確認します。

- さらに、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ないでください。これは、ディスクレスの互換性を維持するためであると同時に、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コマンドの実行を避けるためです。

ファイルの作成や変更、削除に対する要件はさまざまなコマンドで検証できます。たとえば、パッケージが正しく動作するかどうかを `dircmp` や `fssnap` コマンドを使って検証できます。さらに、`ps` コマンドでは、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によってデーモンに対する要件準拠を確認できます。`truss` や `pkgadd` コマンドでは、実行時のパッケージインストール要件に準拠しているかどうかを確認できます。ただし、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または `$BASEDIR` へのアクセスをすべて除外し、読取専用でないディレクトリまたは指定された代替ルート (`/`) 以外のディレクトリへのアクセス情報のみを表示します。

```
# BASEDIR=/a; export BASEDIR
# truss -t open /usr/sbin/pkgadd -R ${BASEDIR} SUNWvxxvm \
2>&1> /dev/null | grep -v O_RDONLY | grep -v \
'open("${BASEDIR}
```

この節で参照したコマンドについては、マニュアルページ `dircmp(1)`、`fssnap(1M)`、`ps(1)`、または `truss(1)` を参照してください。

カスタム JumpStart プログラムのアップグレード要件

パッケージがカスタム JumpStart プログラムの要件に準拠していれば、次に示す従来の Solaris インストールユーティリティを使ってパッケージの追加や削除を行うことができます。

- カスタム JumpStart プログラム
- Solaris 9 `suninstall` プログラム
- Solaris Web Start インストール

さらに、カスタム JumpStart プログラムの要件に準拠したパッケージは Solaris アップグレードに入れることができます。パッケージがカスタム JumpStart プログラムの要件に準拠するためには、代替ルート (/) 要件を満たしていなければなりません。これについては、362 ページの「カスタム JumpStart プログラムと Solaris Live Upgrade の代替ルート (/) 要件」を参照してください。

処理の途中でユーザー入力を求められないほうが、カスタム JumpStart プログラムを効果的に使用することができます。ユーザーの入力作業をなくすためには、`pkgadd` コマンドの `-a` オプションで使用する新しい管理ファイルを設定する必要があります。`-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、パッケージをインストールすることができます。次の例で、`pkgadd` 管理ファイルの使い方を説明します。

- 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。この例では相対管理ファイルの名前が `nocheck` であるため、`pkgadd` は `/var/sadm/install/admin/nocheck` を使用します。

```
# 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/nocheck` を使用します。

```
# pkgadd -a /tmp/nocheck
```

次の例はユーザー定義の管理ファイルの例です。このファイルを使用すると、pkgadd はパッケージをインストールする前に確認のプロンプトを表示しません。

```
mail=  
instance=overwrite  
partial=nocheck  
runlevel=nocheck  
idepend=nocheck  
space=nocheck  
setuid=nocheck  
conflict=nocheck  
action=nocheck  
basedir=default
```

詳細は、マニュアルページ `admin(4)` または `pkgadd(1M)` を参照してください。

Solaris Update リリースへのアップグレード

この付録では、Solaris オペレーティング環境の Update リリースにアップグレードする前にパッチをチェックする手順について説明します。

Solaris Update リリースへのアップグレード

すでに Solaris 9 オペレーティング環境を実行していて、システムに個別のパッチがインストールされている場合、Solaris 9 Update リリースへのアップグレードを行うと、パッチは次のようになります。

- Solaris 9 Update リリースの一部として組み込まれているパッチは自動的に再適用されます。このような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るパッチのうち、Solaris 9 リリースに組み込まれていないパッチは削除されます。

パッチアナライザを使用すれば、Solaris 9 Update リリースにアップグレードしたときに、これらのパッチのうちどれが削除されるかを確認することができます。パッチアナライザの使い方は、インストール方法によって異なります。

- Solaris Web Start を使用してアップグレードする場合、「パッチの解析 (Patch Analyzer)」画面が表示されます。「はい (Yes)」を選択して解析を実行します。
- Solaris `suninstall` プログラムを使用してアップグレードする場合、「パッチの解析 (Patch Analysis)」画面で「解析 (Analyze)」を選択して解析を実行します。
- カスタム JumpStart インストールまたは Solaris Live Upgrade を使用してアップグレードする場合、`analyze_patches` スクリプトを実行して解析を実行します。詳細は、370 ページの「`analyze_patches` スクリプトを実行する方法」を参照してください。

パッチ解析結果の見方については、370 ページの「パッチアナライザの出力の確認方法」を参照してください。

▼ analyze_patches スクリプトを実行する方法

注 - analyze_patches スクリプトを実行するには、NFS またはローカルにマウントされたメディア上にあるスクリプトを使用して、インストールされた既存のシステム、および Solaris 9 DVD、Solaris 9 SOFTWARE CD、またはネットワーク上のイメージにアクセスできる必要があります。

1. Misc ディレクトリに移動します。

- イメージがローカルにマウントされたメディア上にある場合は、次のように入力します。

```
# cd /cdrom/sol_9_Update_sparc/s9/Solaris_9/Misc
```

このコマンドにおいて、*Update* は実際の Update 識別子を表します (例: u1)。

- イメージが NFS ファイルシステムにある場合は、次のように入力します。

```
# cd /NFS_mount_directory/Solaris_9/Misc
```

2. analyze_patches スクリプトを実行します。

```
# ./analyze_patches [-R rootdir] [-N netdir] [-D databasedir]
```

-R <i>rootdir</i>	<i>rootdir</i> には、インストールされてるシステムのルートを指定します。デフォルトで / に設定されます。
-N <i>netdir</i>	<i>netdir</i> 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。 <i>netdir</i> は、solaris_9 ディレクトリを含むディレクトリへのパスです。patch_analyzer を NFS マウントポイントから実行する場合は、このオプションが必要です。
-D <i>databasedir</i>	スクリプトを OS イメージの Misc ディレクトリ以外のディレクトリから起動すると、パッチ解析に使用するデータベースを検索できません。したがって、-D オプションでこのデータベースへのパスを指定します。このデータベースがないと、スクリプトは正しく動作しません。このデータベースは、OS イメージの Solaris_9/Misc/database にあります。

▼ パッチアナライザの出力の確認方法

解析を実行したら、次の手順で出力を検討します。

1. パッチアナライザの出力を確認します。

パッチアナライザからは、ほかのパッチによって削除、ダウングレード、置き換えられ削除されるパッチのリストが出力されます。パッチの置き換えは、パッチアップグレードと同等のものです。置き換えられるパッチは削除され、新しいパッチによってその修正が適用されます。パッチ解析の結果、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.

パッチアナライザからリストが出力されない場合は、システムにすでにインストールされているパッチに対する処理は何も行われません。

2. パッチの置き換えや削除が適切かどうかを判断します。

- 適切な場合は、システムをアップグレードします。
- 適切でない場合は、システムのアップグレードを中止します。

Update リリースでは、アップグレードのかわりに Solaris 9 Maintenance Update リリースを用いて、パッチだけをシステムに適用できます。

注 – Solaris 9 Maintenance Update は CD またはオンラインから入手できます。パッチの適用手順については、『*Maintenance Update Installation Guide*』を参照してください。

用語集

begin 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、Solaris ソフトウェアがシステムにインストールされる前に作業を実行する。このスクリプトは、カスタム JumpStart インストールでのみ使用できる。
DHCP	動的ホスト構成プロトコルは、アプリケーション層プロトコルであり、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する。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する。
disc	磁気ディスク (disk) に対する光学式ディスク。CD (コンパクトディスク) 業界では共通の綴りを使用する。たとえば、CD-ROM は光学式ディスク。
/etc	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	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る。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、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る。ディスクレスクライアントは、起動と実行の際に OS サーバー上の /export ファイルシステムに依存する
finish 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、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作業を実行する。このスクリプトは、カスタム JumpStart インストールでのみ使用できる。
HTTP	(Hypertext Transfer Protocol の略) リモートホストからハイパーテキストオブジェクトをフェッチするインターネットプロトコル。このプロトコルは TCP/IP に基づいている。

IP アドレス

インターネットプロトコル (Internet Protocol, IP) アドレス。TCP/IP では、ネットワーク上の個々のホストを識別する 32 ビットの一意の数値。IP アドレスは、4 つの数をピリオドで区切った形式になる (例: 192.9.9.1)。通常、IP アドレスの各部分は 0~255 の番号であるが、最初の番号は 224 未満とし、最後の番号は 0 以外にする必要がある。

IP アドレスは論理的には、ネットワーク (市外局番のようなもの) とネットワーク上のシステム (電話番号のようなもの) の 2 つの部分に分割される。たとえば、クラス A の IP アドレスは「network.local.local.local」となり、クラス C の IP アドレスは「network.network.network.local」となる。

クラス	範囲 (xxx は 0 から 255 までの数字)	使用できる IP アドレス数
クラス A	1.xxx.xxx.xxx - 126.xxx.xxx.xxx	1,600 万以上
クラス B	128.0.xxx.xxx - 191.255.xxx.xxx	65,000 以上
クラス C	192.0.0.xxx - 223.255.255.xxx	256

IPv6

IPv6 は、現在のバージョン IPv4 (バージョン 4) から拡張されたインターネットプロトコル (IP) の新しいバージョン (バージョン 6) である。IPv6 は IPv4 への追加となる。定義された移行方法を使用して IPv6 を採用すると、現在の運用を中断する必要はない。また、IPv6 には、新しいインターネット機能用のプラットフォームも用意されている。

IPv6 の詳細は、『Solaris のシステム管理 (IP サービス)』を参照。

JumpStart ディレクトリ

カスタム JumpStart インストールの実行に必要なファイルが含まれているディレクトリ。プロファイルフロッピーディスクを使用してインストールする場合は、フロッピーディスク上のルートディレクトリが JumpStart ディレクトリとなる。カスタム JumpStart インストール用にプロファイルサーバーを使用する場合、必要なカスタム JumpStart ファイルをすべて格納するサーバー上のディレクトリが JumpStart ディレクトリとなる。

JumpStart インストール

インストール方法の 1 つ。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、Solaris ソフトウェアをシステムに自動インストールできる。

Kerberos

強力な秘密鍵暗号方式を使用して、クライアントとサーバーが、セキュリティ保護されていないネットワーク接続で相互を認識できるようにするネットワーク認証プロトコル。

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る。

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。
/opt	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
OS サーバー	ネットワーク上のシステムにサービスを提供するシステム。ディスクレスクライアントにサービスを提供するには、OS サーバーは、各ディスクレスクライアントごとに、ルートファイルシステムとスワップ領域 (/export/root、/export/swap) 用のディスク空間が必要である。
rules ファイル	自動的にインストールしたい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する。rules ファイルは、各グループをプロファイルにリンクする。Solaris 9 ソフトウェアをどのようにしてグループ内の個々の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。「プロファイル」も参照。
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する。check スクリプトを使用して rules.ok ファイルを作成しなくてはならない。
Solaris 9 CD イメージ	システムにインストールされた Solaris 9 ソフトウェア。Solaris CD、または Solaris 9 CD イメージからコピーしたインストールサーバーのハードディスク上から利用できる。
Solaris 9 suninstall プログラム	システムを設定し、そのシステムに Solaris 9 ソフトウェアをインストールするために使用できるコマンド行インタフェース (CLI) で、メニューから選択可能な対話式のスクリプト。
Solaris Web Start	グラフィカルユーザーインタフェース (GUI) とコマンド行インタフェース (CLI) を備えたインストールプログラム。ウィザードパネルに、Solaris 9 ソフトウェアやサードパーティソフトウェアをインストールする手順が示される。
sysidcfg ファイル	システムを事前設定する特殊な一連のシステム構成キーワードを指定するファイル。
/usr	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収められている。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 Solaris 9 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ディスク容量を最小限に抑えることができる。
/var	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 ファイルなどがある。

アーカイブ	マスターシステムからコピーされたファイルがすべて入ったファイル。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る。アーカイブをシステムにインストールすると、マスターシステムとまったく同じ構成でインストールされる。
アクティブブート環境	現在ブートされているブート環境。
アップグレードオプション	Solaris Web Start プログラムと Solaris suninstall プログラムによって提示されるオプション。アップグレード時には、新しいバージョンの Solaris とディスク上の既存のファイルが結合される。前回 Solaris をインストールしてから加えたローカルの変更内容をできる限り残す。
インストールサーバー	インストール用に、Solaris 9 CD イメージをネットワーク上のほかのシステムに提供するサーバー(「メディアサーバー」とも呼ばれる)。Solaris 9 CD 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る。
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る。
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
カスタム JumpStart	ユーザーが定義したプロファイルに基づいて、Solaris ソフトウェアをシステムに自動的にインストールする方法。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る。カスタム JumpStart インストールは、ユーザーが作成する JumpStart インストールである。
カスタムプローブファイル	rules ファイルと同じ JumpStart ディレクトリに存在しなければならないファイルで、プローブと比較という 2 つのタイプの関数を含む Bourne シェルスクリプト。プローブ関数は作業を実行して、必要な情報を収集したり、定義に対応した SI_ 環境変数を設定する。プローブ関数は、プローブキーワードになる。比較関数は、対応するプローブ関数を呼び出してプローブ関数の出力を比較し、キーワードが一致する場合は 0、キーワードが一致しない場合は 1 を返す。比較関数はルールキーワードになる。「rules ファイル」を参照。
共有可能ファイルシステム	/export/home や /swap のようなユーザー定義のファイルシステム。Solaris Live Upgrade の使用時に、アクティブブート環境と非アクティブブート環境によって共有される。共有可能ファイルシステムは、アクティブブート環境の vfstab 内と非アクティブブート環境の vfstab 内に同じマウントポイントを持つ。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る。共有可能ファイルシステムはデフォルトで共

	有されるが、ユーザーが宛先スライスを指定することもできる。この場合、そのファイルシステムがコピーされる。
クライアント	通信用のクライアントサーバーモデルでは、計算機能や大容量のメモリといったサーバーの資源にリモートアクセスするプロセスがクライアントに相当する。
クラスタ	パッケージ(ソフトウェアモジュール)を論理的に集めたもの。Solaris 9ソフトウェアは4つの「ソフトウェアグループ」に分割され、それぞれがクラスタと「パッケージ」から構成されている。
クリティカルファイルシステム	Solaris オペレーティング環境が必要とするファイルシステム。Solaris 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それぞれの <code>vfstab</code> では独立したマウントポイントになる。 <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ある。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る。
クローンシステム	フラッシュアーカイブを使ってインストールされたシステム。クローンシステムは、マスターシステムと同一のインストール構成になる。
コアシステムサポート	システムで Solaris オペレーティング環境を起動して実行するの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。これ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るが、CDE ソフトウェアは含まれない。
コマンド行	コマンドで始まる文字列。多くの場合、コマンドの後には引数 (オプション、ファイル名、式などの文字列) が続き、行末 (EOL) 文字で終わる。
サーバー	資源を管理し、クライアントにサービスを提供するネットワークデバイス。
サブネット	単一の論理ネットワークをより小さな物理ネットワークに分割して、経路指定を簡素化する方式。
サブネットマスク	サブネットアドレス指定のため、インターネットアドレスからビットを選択するために使用されるビットマスク。マスクは32ビット。インターネットアドレスのネットワーク部分と、ローカル部分の1個以上のビットを選択する。
時間帯	グリニッジ標準時間を基準に地球の表面を24の地域に経度分割したものの。
初期インストールオプション	Solaris Web Start プログラムおよび Solaris <code>suninstall</code> プログラムの実行時に提示される、新バージョンの Solaris でディスクを上書きするオプション。初期インストールオプションはアップグレード可能なシステムでも表示されるが、初期インストールオプションを選択した場合、旧バージョンの Solaris ソフトウェア (ローカルの変更内容も含めて) が含まれているディスクは上書きされる。

ジョブ	コンピュータシステムで実行されるユーザー定義の処理。
スタンドアロン	ほかのマシンからのサポートを一切必要としないコンピュータ。
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。
スワップ領域	システムに現在のプロセスを処理できるだけの十分なシステムメモリーがない場合に、仮想記憶領域として使用されるディスク空間。/swap または swap ファイルシステムともいいます。
全体ディストリビューション	Solaris 9 の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	Solaris 9 のリリース全体と、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。Solaris を SPARC サーバーシステム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する。
ソフトウェアグループ	Solaris ソフトウェアの論理グループ(クラスとパッケージ)。Solaris のインストール時には、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、全体ディストリビューションおよび OEM サポートをインストールできる。
代替ルート (0) ファイルシステム	現在実行中のシステムではなくオペレーティング環境のコピー。
チェックサム	一連のデータ項目を合計した結果。一連のデータ項目を検査するために使用される。データ項目は、数値でも、文字列でもよく、文字列の場合はチェックサム計算時に数値として扱われる。チェックサムの値から、2つのデバイス間の情報交換が正しく行われたかを確認できる。
ディスク	1枚以上の磁性体の円盤から成るメディアであり、ファイルなどのデータを格納する同心トラックとセクターで構成される。「disc」も参照。
ディスク構成ファイル	ディスクの構造(たとえば、バイト/セクター、フラグ、スライス)を表現するファイル。ディスク構成ファイルにより、単一システムから pfinstall を使用して、サイズの異なるディスクのプロファイル进行测试できる。
ディスクレスクライアント	ディスク記憶装置を持たないためサーバーに依存するクライアント。
電源管理システム	<p>システムの状態を自動的に保存し、30分間アイドル状態が続くと電源を切断するソフトウェア。米国環境保護庁の省電力 (Energy Star) ガイドライン第2版に準拠したシステム (sun4u SPARC システムなど) に Solaris ソフトウェアをインストールすると、デフォルトで電源管理ソフトウェアがインストールされる。リブート後、電源管理ソフトウェアを有効にするかどうかを確認するメッセージが表示される。</p> <p>Energy Star ガイドラインでは、システムまたはモニターを使用していない場合は、自動的に「休眠状態」(30ワット以下の消費)に入ることが要求される。</p>

動的プロファイル	カスタム JumpStart インストール時に、begin スクリプトによって動的に作成されるプロファイル
ドメイン	インターネットのネーミング階層の一部。ドメインは管理ファイルを共有する、ローカルネットワーク上のシステムグループを表すドメイン。
ドメイン名	ローカルネットワーク上のシステムグループに割り当てられた名前であり、管理ファイルを共有する。ネットワーク情報サービス (NIS) のデータベースが正常に動作するためにはドメイン名が必要。ドメイン名は、ピリオドで区切られた一連の構成要素名から構成される (たとえば、tundra.mpk.ca.us)。ピリオドで区切られた各構成要素名は右側に行くにしたがって、全体的な (リモートな) 管理権限領域を表す。
ネームサーバー	ネットワーク上のシステムに対してネームサービスを提供するサーバー。
ネームサービス	ネットワーク上の全システムに関する重要なシステム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。ネットワーク上のシステムは、これを利用して相互通信を行う。ネームサービスを使用することによって、ネットワーク全域にわたるシステム情報を保守、管理、または取得できる。Sun では NIS と NIS+ のネームサービスをサポートしている。ネームサービスを使用しないと、各システムは (ローカルの /etc ファイルに) システム情報のコピーを保持しなければならない。
ネットワークに接続されているシステム	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。ローカルエリアネットワーク (LAN) とも呼ばれる。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。
ネットワークインストール	(CD-ROMドライブがあるシステムからCD-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。
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。
ハードリンク	ディスク上のファイルを参照するディレクトリエントリ。複数のハードリンクから同じ物理ファイルを参照することができる。
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。Solaris ソフトウェアは 4 つの「ソフトウェアグループ」に分類され、それぞれが「クラスタ」とパッケージで構成される。
パッチアナライザ	手作業でも、Solaris suninstall プログラムや Solaris Web Start プログラム内でも実行できるスクリプト。パッチアナライザは、システムを解析し、Solaris 9 Update へのアップグレードを行うことで削除されるパッチがどれであるかを判断します。

非アクティブブート環境	現在ブートされておらず、次のリブートでのアクティブ化もスケジュールされていないブート環境。
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	SunOS™ オペレーティングシステムでは、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のこと。
ファンクションキー	F1、F2、F3 などの名前が付いた 10 個以上のキーボードキー。これらのキーにはそれぞれ特定の機能が割り当てられている。
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。
ブート環境	1 つ以上のディスクスライス、適切なマウントポイント、ファイルシステムから構成されるブート可能な Solaris 環境。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る。
ブートサーバー	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の存在するサブネットが、Solaris ソフトウェアをインストールする必要があ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。
フォーマット	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。
フォールバック	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う。
フラッシュ	マスターシステムと呼ぶシステムからファイルのアーカイブを作成する 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る。
プラットフォームグループ	特定のソフトウェア用にベンダーが定義するハードウェアプラットフォームのグループ。たとえば i86pc や sun4c など。
プラットフォーム名	uname -i コマンドによって出力される情報。たとえば Ultra 60 のプラットフォーム名は、SUNW,Ultra-60。
プロンプトキーワード	ルールに必要な照合条件を設定してプロファイルを実行せずに、システムに関する属性情報を抽出するための構文要素。「ルール」も参照。
プロファイル	Solaris ソフトウェアのシステムへのインストール方法 (たとえば、どのソフトウェアグループをインストールするか) を定義するテキストファイル。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する。通常

は、ルールごとに異なるプロファイルを作成するが、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。「rules ファイル」を参照。

プロファイルサーバー	すべての重要なカスタム JumpStart ファイルを JumpStart ディレクトリに持つサーバー。
プロファイルフロッピーディスク	すべての重要なカスタム JumpStart ファイルを、そのルートディレクトリ (JumpStart ディレクトリ) に持つフロッピーディスク。
ホスト名	システムがネットワーク上のほかのシステムから識別される名前。この名前は、特定のドメイン (通常、これは 1 つの組織内にあることを意味する) 内にある全システム間で固有でなければならない。ホスト名は、文字、数字、マイナス符号 (-) を任意に組み合わせて作成できるが、先頭と末尾にマイナス符号は使用できない。
ボリューム	連結、ストライプ化、ミラー化、セットアップ用 RAID5 ボリューム、ロギング用物理デバイスによって単一の論理デバイスとしてアクセスされる物理スライスのグループ。作成されたボリュームは、スライスと同じように使用できる。ボリュームは、1 台の物理デバイス上の正しい場所に論理ブロックアドレスをマップする。マップの種類は、個々のボリュームの構成によって決まる。 標準の UNIX 用語では、擬似デバイスまたは仮想デバイスとも呼ばれる。
ボリュームマネージャ	CD-ROM や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手段を提供するプログラム。
マウント	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 /usr) が必要。
マウント解除	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。
マウントポイント	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。
マスターシステム	フラッシュアーカイブを作成するシステム。このシステム構成がアーカイブに保存される。
ミニルート	起動可能な最小の Solaris ルートファイルシステム。ミニルートには、カーネルと、Solaris 環境をハードディスクにインストールするために必要な最小限のソフトウェアが含まれる。ミニルートは、初期インストールでマシンにコピーされるファイルシステム。
メタデバイス	「ボリューム」を参照。
メディアサーバー	「インストールサーバー」を参照。
矢印キー	数値キーパッド上にある方向を示す 4 つのキーの 1 つ。

ユーティリティ	コンピュータを購入すると通常無料で提供される、標準プログラム。
/ (ルート)	複数の項目から成る階層構造において、ほかのすべての項目を子孫として持つ唯一の項目を指す。階層内において、ルート項目より上位の項目は存在しない。/ は、直接的または間接的にその他のすべてのディレクトリの元となるベースディレクトリ。ルートディレクトリには、カーネル、デバイスドライバ、システムの起動 (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る。
ルール	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
ロケール	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 en_US、英国英語のロケールは en_UK。

索引

数字・記号

#

- rules ファイル内の, 152
- プロファイル内の, 156

A

add_install_client コマンド

JumpStart ディレクトリのアクセス, 148

インストールサーバーの設定

- CD メディアによる, 88
- DVD メディアを使用する, 76

説明, 91

例

- DHCP による (CD メディア), 90
- DHCP を使用する、DVD メディアを使用する, 78
- 同じサブネット (CD メディア), 90
- 同じサブネット、DVD メディアを使用する, 78
- ブートサーバー、DVD メディア, 78
- ブートサーバーの (CD メディア), 90

add_to_install_server、説明, 91

AND ルールフィールド, 153

any

プローブキーワード

説明と値, 230

ルールキーワード, 197, 229

archive_location キーワード, 204

arch プローブキーワード, 229

arch ルールキーワード, 197, 229

auto_install_sample ディレクトリ

check スクリプト, 163, 181

JumpStart ディレクトリへファイルをコピー,
151

set_root_pw finish スクリプト, 170, 171

ファイルを JumpStart ディレクトリにコピー,
147

B

backup_media キーワード, 209

banner コマンド, 92

begin.log ファイル, 165

begin スクリプト

rule field, 153

アクセス権, 166

概要, 165

サイト固有のインストールプログラム, 175

動的プロファイルの作成, 166, 167

begin ルールフィールドの説明, 153

boot: cannot open /kernel/unix メッセージ,
347

boot_device キーワード, 210

bootparams ファイル

JumpStart ディレクトリへのアクセスを有効
にする, 148

更新, 350

Bourne シェルスクリプト、ルールフィールド
内の, 153

C

- c option, add_install_client コマンド, 89
- cannot open /kernel/unix メッセージ, 347
- Can't boot from file/device メッセージ, 347
- CHANGE DEFAULT BOOT DEVICE メッセージ, 350
- check スクリプト
 - custom_probes.ok ファイルの作成, 181
 - custom_probes ファイルの妥当性検査, 181, 182
 - rules ファイルの作成, 163
 - rules の妥当性検査, 164
 - rules ファイルの妥当性検査, 182
 - rules ファイルの妥当性検査, 163, 164
 - 動的プロファイルと, 167
 - ルールの検査, 182
- check スクリプトの -r オプション, 182
- client_arch キーワード, 211
- client_name, 説明, 89
- client_root プロファイルキーワード, 212
- clock gained xxx days メッセージ, 347
- cluster プロファイルキーワード
 - 説明と値, 213
 - 例, 157
- CPU (プロセッサ)
 - プローブキーワード, 229
 - ルールキーワード, 197, 229
- .cshrc ファイル, 169
- custom_probes.ok ファイル
 - 作成, 181
- custom_probes.ok ファイル, 作成, 181
- custom_probes.ok ファイル
 - 説明, 181
- custom_probes ファイル
 - check による妥当性検査, 181
 - check を使用する妥当性検査, 182
 - custom_probes のテスト, 182
- custom_probes ファイル
 - 名前の指定, 178
 - 要件, 178
- c オプション
 - add_install_client コマンド, 196
 - pfinstall コマンド, 162

D

- default, partitioning, 223

- Developer Solaris Software Group, プロファイル例, 157
- dfstab ファイル, 146, 193
- DHCP (Dynamic Host Configuration Protocol) の事前設定, 48
- disksize ルールキーワード
 - 説明と値, 198, 229
- disks プロブキーワード, 説明と値, 229
- domainname プロブキーワード, 229
- domainname ルールキーワード, 198, 229
- dontuse プロファイルキーワード, 214, 226
- d オプション, add_install_client コマンド, 89

E

- eng_profile の例, 193
- /etc/bootparams ファイル
 - JumpStart ディレクトリへのアクセスを有効にする, 148, 350
- /etc/dfs/dfstab ファイル, 146, 193
- /etc/locale ファイル, 55
- /etc/mnttab ファイル, 150
- existing partitioning の値, 223
- explicit partitioning の値, 223
- /export ファイルシステム, 33

F

- fdformat コマンド, 150
- filesys キーワード, 215
- filesys プロファイルキーワード
 - 説明と値, 214
 - 例, 157
- finish.log ファイル, 167
- finish スクリプト
 - システムの root パスワードを設定する, 170
 - パッケージとパッチの追加, 168
 - ファイルの追加, 167
 - ルート環境のカスタマイズ, 169
 - ルールフィールド, 154
- finish ルールフィールドの説明, 154
- flarcreate コマンド, 128
- flar コマンド, 131

G

geo キーワード, 217
getfile: RPC failed: error 5: RPC Timed out メッセージ, 149

H

hostaddress プロローブキーワード, 229
hostaddress ルールキーワード, 198, 229
hostname プロローブキーワード, 説明と値, 229
hostname ルールキーワード
説明と値, 199, 229
例, 197

I

install_config コマンド, 148, 149
install_type キーワード, 219
install_type プロファイルキーワード
プロファイルのテスト, 160, 162
要件, 156, 157
例, 157
installed プロローブキーワード, 説明と値, 229
installed ルールキーワード
説明と値, 199, 229
IPv6 の事前設定, 48
IP アドレス
事前設定, 48
プロローブキーワード, 229
ルールキーワード, 198, 229
isa_bits キーワード, 219

J

JumpStart ディレクトリ
finish スクリプトによるファイルの追加,
167, 168
rules ファイルの例, 152
アクセス権, 145, 149
共有, 145, 193
作成, 192
サーバー, 145
フロッピーディスク, 150
ファイルのコピー, 168
インストールファイル, 147

ファイルのコピー (続き)
ファイルのコピー
インストールファイル, 151
JumpStart ディレクトリの共有, 193

K

karch プロローブキーワード, 229
karch ルールキーワード, 199, 229
Kerberos
構成情報, 39
事前設定, 48

L

layout_constraint キーワード, 219
le0: No carrier - transceiver cable problem メッセージ, 347
Live Upgrade
Solaris Live Upgradeを参照
Live Upgrade ブート環境の名前を表示する,
307
Live Upgrade ブート環境を削除する, 306
Live Upgrade ブート環境を比較する, 304
locale.org_dir テーブルエントリの追加, 56
locale キーワード, 221

M

Makefile ファイル, 54
marketing_profile の例, 194
memsize プロローブキーワード, 説明と値, 229
memsize ルールキーワード
説明と値, 200, 229
mnttab ファイル, 150
model プロローブキーワード, 説明と値, 229
model ルールキーワード
説明と値, 200, 229
modify_install_server, 説明, 91
mount コマンド, 91

N

netmask の事前設定, 48

network プロープキーワード, 説明と値, 229
network ルールキーワード
説明と値, 200, 229
nistbladm コマンド, 56
No carrier - transceiver cable problem メッセージ, 347
Not a UFS filesystem メッセージ, 347
num_clients プロファイルキーワード, 222

O

osname プロープキーワード, 230
osname ルールキーワード, 201, 230

P

package プロファイルキーワード, 説明と値, 222
partitioning
使用しないディスク, 214
プロファイルキーワード, 223, 226
partitioning キーワード, 223
pfinstall コマンド, 159
probe ルールキーワード, 説明と値, 201
profile キーワード, backup_media, 204
prtvtoc コマンド, ディスク構成ファイルの作成, 173
-p オプション, check スクリプトの, 164, 182

R

reset コマンド, 92
root_device キーワード, 224
rootdisk
filesystem のスライス値, 216
JumpStart で設定される値, 224
定義, 224
root パスワード, 事前設定, 48
root パスワードを finish スクリプトで設定する, 170
RPC failed: error 5: RPC Timed out メッセージ, 149
RPC Timed out メッセージ, 149
rule_keyword ルールフィールド, 153
rule_value ルールフィールド, 153

rules, フィールドの説明, 154
rules.ok ファイル
作成, 163
説明, 163
ルールの照合順序, 154, 183
rules ファイル
check による妥当性検査, 164
check を使用する妥当性検査
動的プロファイルと, 167
check を使った妥当性検査
カスタム JumpStart の例, 195
rules の妥当性検査, 164
カスタム JumpStart の例, 194, 195
構文, 153
コメント, 152
作成, 152
説明, 152
名前, 153
名前の指定, 152
複数行のルール, 152
ルールの追加, 153
例, 152
-r オプション, check スクリプトの, 164

S

scripts, ルールフィールド内の Bourne シェルスクリプト, 153
set_root_pw finish スクリプト, 170, 171
setup_install_server, 説明, 91
setup_install_server コマンドの -b オプション, 192
shareall コマンド, 146, 193
share コマンド, JumpStart ディレクトリの共有, 193
SI_CONFIG_DIR 変数, 167
SI_PROFILE 環境変数, 166
size, ローカルファイルシステム, 216
slices
filesystem の値, 215
プロープキーワード, 229
ルールキーワード, 199
Solaris Live Upgrade
アップグレード障害の回復, 294
アップグレードの作業マップ, 280
インストール, 257
概要, 234

- Solaris Live Upgrade (続き)
 - 起動, 258
 - コピー
 - 共有可能なファイルシステム, 275
 - コマンド, 323
 - 処理を取り消す, 304
 - 停止, 258
 - ファイルシステムの構成, 261
 - ファイルシステムのスライス, 262
 - ファイルへ出力, 262
 - ブート環境のアクティブ化, 291
 - ブート環境のアップグレード, 280
 - ブート環境の構成の表示, 312
 - ブート環境の削除, 306
 - ブート環境の作成, 234, 259
 - ブート環境の名前の表示, 307
 - ブート環境の名前の変更, 308
 - ブート環境を比較する, 304
 - フラッシュアーカイブのインストール, 286
 - Solaris Web Start
 - インストールプログラム, 93
 - グラフィカルユーザーインタフェース (GUI), 93
 - コマンド行インタフェース (CUI), 93
 - 実行方法, 93
 - フラッシュアーカイブのインストール, 120
 - ほかのインストール方法との比較, 27
 - Solaris Live Upgrade 処理を取り消す, 304
 - Solaris Live Upgrade のコマンド, 323
 - Solaris Live Upgrade ブート環境の名前を変更する, 308
 - Solaris Live Upgrade 用にブート環境のアクティブ化、概要, 241
 - Solaris Live Upgrade 用のブート環境のアクティブ化の失敗、概要, 243
 - Solaris Live Upgrade 用のブート環境のブートの失敗、概要, 243
 - Solaris suninstall プログラム
 - 実行方法, 101
 - フラッシュアーカイブのインストール, 121
 - Solaris 9 ソフトウェア
 - グループ, 213, 214
 - アップグレード, 214
 - パッケージの指定, 222
 - プロファイル例, 157
 - リリースまたはバージョン
 - installed プロンプキーワード, 229
 - Solaris 9 ソフトウェアのバージョン, installed
 - ルールキーワード, 199
 - Solaris 9 ソフトウェアのリリース
 - installed プロンプキーワード, 229
 - installed ルールキーワード, 199, 229
 - osname プロンプキーワード, 230
 - osname ルールキーワード, 201, 230
 - Solaris ソフトウェア
 - リリースまたはバージョン, 201, 230
 - Solaris ソフトウェアのバージョン
 - installed プロンプキーワード, 229
 - installed ルールキーワード, 229
 - osname プロンプキーワード, 230
 - osname ルールキーワード, 201, 230
 - stty コマンド, 37, 42, 185
 - SUNWCall グループ, 213
 - SUNWCprog グループ, 213
 - SUNWCreq グループ, 213
 - SUNWCuser グループ, 213
 - SUNWCXall グループ, 213
 - swap ファイルシステム
 - サイズの決定, 212
 - メモリーのサイズと, 212
 - sysidcfg ファイル, ガイドラインと要件, 49
 - sysidcfg ファイル
 - キーワード, 50
 - 構文, 50
 - 作成方法, 53
 - system_type プロファイルキーワード
 - 説明と値, 225
 - 例, 157
- ## T
- tftpd デーモン, 350
 - tip ライン接続とカスタム JumpStart インストール, 184
 - totaldisk プロンプキーワード, 230
 - totaldisk ルールキーワード, 202, 230
 - transceiver cable problem メッセージ, 347
- ## U
- UFS, 150
 - uname コマンド, 92
 - Unknown client error メッセージ, 345

upgrade_log ファイル, 98, 104
usedisk プロファイルキーワード, 説明と値,
226

V

/var/sadm/begin.log ファイル, 165
/var/sadm/finish.log ファイル, 167
/var/sadm/install_data/upgrade_log ファイル
, 104
/var/yp/Makefile, 54
/var/yp/make コマンド, 56
volcheck コマンド, 150

W

WARNING: CHANGE DEFAULT BOOT
DEVICE, 350
WARNING: clock gained xxx days メッセージ,
347
Web Start Flash
キーワード
カスタム JumpStart, 204
説明, 107

あ

アーカイブ
Web Start Flashを参照
アクセス権
begin スクリプト, 166
finish スクリプト, 167
JumpStart ディレクトリ, 145, 149
新しい行に続ける、rules ファイル内で, 152
アップグレード
Solaris Live Upgrade による, 280
Solaris Live Upgrade による、概要, 239
Solaris Update リリースへ, 369
カスタム JumpStart による, 60
作業の概要, 21
システム要件, 24
初期インストールとの比較, 23, 60
ディスク容量の再配置, 61
ディスク容量の推奨, 33
方法, 24, 60

アップグレード (続き)
方法の比較, 27
アップグレードインストール
Solaris Live Upgrade のためのアップグレー
ド障害の回復, 294
アップグレードの失敗, 352
カスタム JumpStart インストール, 183
プロファイルキーワード, 214, 219, 222
ログファイル, 98, 104
アップグレード後の整理, 98, 104
アップグレード障害, Solaris Live Upgrade によ
る回復, 294
アップグレードの失敗
リブート時の問題, 352
リブートの問題, 352
アップグレードのための準備
アップグレードする前に必要な情報, 43
システムの準備, 42
アンパサンド (&&) ルールフィールド, 153

い

インストール
アップグレードとの比較, 23
作業の概要, 21
システム要件, 24
ディスク容量の推奨, 33
方法の比較, 27
インストールサーバー
CD メディアによる作成, 81
CD メディアによる作成、例, 85
CD メディアによるネットワークインストー
ル設定, 88
DVD メディアを使用する作成, 71
DVD メディアを使用する作成の例, 73
DVD メディアを使用するネットワークイン
ストールのための設定, 76
サブネット上の, 73, 85
適切なシステムの種類, 65
インストールの準備
カスタム JumpStart インストール, 142
システム構成情報の事前設定
方法, 48
利点, 47
インストールのための準備
インストールする前に必要な情報, 38
システムの準備, 37

え

- エンドユーザーシステムサポート, 213
 - 説明, 34
- 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, サイズ, 35

か

- 開始, check スクリプト, 164
- 開発者システムサポート, 213
 - 説明, 34
- 開発者システムサポートソフトウェアグループ, サイズ, 35
- カスタム JumpStart インストール
 - tip ライン接続, 184
 - いつアップグレードするか, 60
 - オプション機能
 - begin スクリプト, 167
 - finish スクリプト, 171
 - 概要, 141
 - 準備, 142, 164
 - 説明, 142
 - 他のインストール方法との比較, 28
 - ブートとインストール, 183
 - フラッシュアーカイブのインストール, 122
- 例
 - check スクリプト, 195
 - JumpStart ディレクトリ, 193
- 代わりのインストールプログラム, 175
- !(感嘆符) ルールフィールド, 153
- 感嘆符 (!) ルールフィールド, 153

き

- キーワード
 - sysidcfg ファイル, 50
 - カスタム JumpStart
 - Web Start Flash アーカイブ, 204
 - プローブ, 177
- 起動, tftpd デーモン, 350
- 共有, JumpStart ディレクトリの, 146
- 共有可能なファイルシステム、定義, 234
- 共有コマンド, JumpStart ディレクトリの共有, 146

く

- クラスタのプロファイルキーワード, 説明と値, 214
- クリティカルなファイルシステム、定義, 234

け

- 計画
 - Live Upgrade のシステム要件, 247
 - Solaris Live Upgrade, 247
 - インストール方法の比較, 27
 - 作業の概要, 21
 - システム要件, 24
 - 初期インストールとアップグレードの比較, 23
 - ディスク容量, 33
 - ネットワークからインストール, 25
- 検査
 - rules ファイルの妥当性検査, 164
 - プロファイル, 160
- 検証, プロファイル, 160

こ

- コアシステムサポート, 213
- 構成
 - Solaris Live Upgrade ファイルシステム, 261
 - ディスク構成ファイルの作成, 173
- コピー
 - JumpStart インストールファイル, 147, 151
 - JumpStart ディレクトリファイル, 168
 - Solaris Live Upgrade ファイルシステム, 302
- コメント
 - rules ファイル内の, 152
 - プロファイル内の, 156

さ

- サーバー
 - CD メディアによるネットワークインストールの設定
 - スタンドアロンインストール, 88

CD メディアによるネットワークインストールの設定 (続き)

DVD メディアを使用するネットワークインストールの設定

スタンドアロンインストール, 76

JumpStart ディレクトリの作成, 145

ネットワークインストールの要件, 65

ルート領域, 212

サイズ

tip ウィンドウの大きさ, 184

スワップ空間

最大サイズ, 212

プロファイルの例, 142

スワップ領域

ディスクレスクライアント, 212

ハードディスク, 72, 82, 87, 198, 202, 229, 230

プロンプトキーワード, 229

ルート領域, 212

メモリー, 200, 229

再設定する、入出力割り込みの後にディスプレイと端末を, 92

サイト固有のインストールプログラム, 175

削除

クラスタ、アップグレード時の, 214

ソフトウェアグループからパッケージを, 222

作成

CD メディアによるインストールサーバーの, 81

CD メディアを使用してインストールサーバーを, 80

custom_probes.ok ファイル, 181

custom_probes.ok ファイル, 181

DVD メディアを使用するインストールサーバー, 71

DVD メディアを使用するインストールサーバーの, 69

DVD メディアを使用するサブネット上でのブートサーバーの, 74

/etc/locale ファイル, 55

JumpStart ディレクトリ

サーバー, 145

Live Upgrade のブート環境, 234, 259

rules.ok ファイル, 181

rules.ok ファイル, 163

rules ファイル, 152

sysidcfg ファイル, 53

作成 (続き)

UFS, 150

サブネット上でブートサーバーを

CD メディアを使用して, 86

サブネット上にブートサーバーを作業、CD メディア, 80

サブネット上のブートサーバー作業、DVD メディア, 70

ディスク構成ファイル, 173

プロファイル, 166

説明, 155

ローカルファイルシステムの, 215

サブネット

CD メディアによるブートサーバー作成, 86

DVD メディアを使用するブートサーバーの作成, 74

し

時間帯の事前設定, 48

試験、プロファイル, 61

システム構成情報の事前設定

NIS+ による言語またはロケール, 56

NIS による言語またはロケール, 54

sysidcfg ファイルを使用する, 48

電源管理, 57

ネームサービスによる, 48, 54

方法の選択, 48

利点, 47

システム情報の表示, 92

出力ファイル

begin スクリプトログ, 165

finish スクリプトログ, 167

アップグレードログ, 98, 104

準備、Solaris 9 をインストールする、カスタム

JumpStart インストール, 164

照合

rootdisk の値, 224

動的プロファイル, 166

ルールでの順序, 154

ルールの順序, 183

す

スクリプト

begin スクリプト, 165, 167, 175

finish スクリプト, 167, 171, 175

スタンドアロンシステム, プロファイル例,
157

スタンドアロンシステム, カスタム JumpStart
インストールの例, 138

ステータス, ブート環境を表示する, 300

スライス

Live Upgrade のガイドライン, 251

Live Upgrade のためにファイルシステムを
カスタマイズ, 262

プロファイル例, 157

ルールキーワード, 229

スワップファイルシステム

Live Upgrade のためのカスタマイズ (キャラ
クタインタフェース), 264

Live Upgrade 用のカスタマイズ (コマンド
行), 271

Solaris Live Upgrade の概要, 235

ディスクレスクライアントのスワップ領域,
212

プロファイルの例, 142

せ

セキュリティ

root パスワード, 170, 171

セキュリティポリシー, 事前設定, 48

全体ディストリビューション, 213

説明, 34

全体ディストリビューションソフトウェアグ
ループ, サイズ, 35

全体ディストリビューションと OEM サポート,
213

説明, 34

全体ディストリビューションと OEM サポート
ソフトウェアグループ, サイズ, 35

そ

ソフトウェアグループ

アップグレード, 24, 60, 214

サイズ, 35

説明, 34

ソフトウェアグループ (続き)

パッケージの指定, 222

プロファイルの, 213, 214

プロファイル例, 157

た

対話式インストールプログラム, ほかのインス
トール方法との比較, 28

妥当性検査

custom_probes ファイル

check を使用する, 182

テスト, 182

rules ファイル

check の使用, 163, 164, 182

カスタム JumpStart の例, 195

動的プロファイルと, 167

端末タイプの事前設定, 48

つ

追加

finish スクリプトによりパッケージとパッチ
を, 168

finish スクリプトによりファイルを, 167

locale.org_dir テーブルエントリ, 56

rules ファイルにルールを, 153

インストールサーバー構成

CD メディアによる, 88

インストールサーバーの構成

DVD メディアを使用する, 76

クラスタ、アップグレード時の, 214

システムをネットワークから, 70

ソフトウェアグループ内のパッケージ, 222

データレスクライアント

DVD メディアを使用する, 76

データレスクライアントを

CD メディアを使用して, 88

ネットワーク上でシステムを, 80

て

ディスク構成ファイル

作成, 173

説明, 159, 173

ディスク容量
計画, 33
ソフトウェアグループの要件, 35

ディスクレスクライアント
スワップ領域, 212
プラットフォーム, 211

ディレクトリ
JumpStart, 147, 149, 152, 167, 168, 192
アクセス権, 145
インストールファイルのコピー, 147
インストールファイルをコピー, 151
システムのために作成, 149
ディレクトリの共有, 146, 193
変更, 163, 181
ローカルディスク上の Solaris 9 ソフトウェアのイメージに, 147, 151

デーモン, 350

テスト
custom_probes ファイルの妥当性検査
check の使用, 181
custom_probes のテスト, 182
rules ファイルの妥当性検査
check の使用, 182
rules ファイルの妥当性検査
check の使用, 163, 164
カスタム JumpStart の例, 195
動的プロファイルと, 167
プロファイル, 159, 162

デフォルト
partitioning
使用しないディスク, 214
ディスクの指定, 226
SI_CONFIG_DIR 変数, 167
インストールされるソフトウェアグループ,
213
動的プロファイル名, 166
デフォルトのルーター, 事前設定, 48
電源管理, 48, 57

と
動的プロファイル, 166, 167

ドメイン
プローブキーワード, 229
ルールキーワード, 198, 229
ドメイン名の事前設定, 48

な
名前/名前の指定
rules ファイル, 152
ソフトウェアグループ, 214
名前/名前の割り当て, ホスト名, 89
名前/命名
動的プロファイル名, 166
ホスト名, 199, 229
名前の指定
custom_probes ファイル, 178
システムプラットフォーム名, 92
名前の割り当て
rules ファイル, 153
システムのモデル名, 200, 229

ね
ネームサーバーの事前設定, 48
ネームサービス, 事前設定, 48
ネットワークインストール
CD メディアによる, 81
DHCP を使用, 67
DVD メディアを使用する, 70, 74
準備, 65
説明, 65
要件, 65
ネットワークインタフェース, 48
ネットワーク上でのインストール, CD メディア
による, 86
ネットワーク上のインストール, カスタム
JumpStart インストール, 141
ネットワーク番号, 200, 229

は
パーティション分割, 例, 157
ハードディスク
partitioning
partitioning default の指定, 226
partitioning デフォルトに含められない,
214
プロファイルキーワード, 223
rootdisk の値, 224
サイズ, 72, 82, 87, 198, 202, 229, 230
プローブキーワード, 229
ルート領域, 212

- サイズ (続き)
 - スワップ空間
 - 最大サイズ, 212
 - プロファイルの例, 142
 - プロファイル例, 157
 - スワップ領域
 - ディスクレスクライアント, 212
 - パーティション分割
 - 例, 157
 - マウント, 214
 - パス
 - check スクリプト, 163, 181
 - パスワード、root の, 170, 171
 - バックスラッシュ、rules ファイル内の, 152
 - パッケージ
 - chroot により追加, 169
 - finish スクリプトによる追加, 168
 - JumpStart プログラムや Solaris Live Upgrade を使用するときの要件, 361
 - Live Upgrade で追加、概要, 282
 - Solaris Live Upgrade 代替ブート環境の要件, 364
 - 管理ファイル, 165
 - 代替ルート (/) に対する要件, 362
 - パッチ
 - chroot により追加, 169
 - finish スクリプトによる追加, 168
 - Live Upgrade で追加、概要, 282
 - Update リリースにアップグレードするときの制限, 24
 - パッチアナライザ, 369
- ひ
- 日付の事前設定, 48
 - 表示
 - tip ライン接続とカスタム JumpStart インストール, 184
 - システム情報, 92
 - プラットフォーム名, 92
 - マウントされたファイルシステム, 91
- ふ
- ファイルとファイルシステム
 - begin スクリプト出力, 165
 - ファイルとファイルシステム (続き)
 - finish スクリプト出力, 167
 - Live Upgrade におけるファイルシステムの共有、概要, 253
 - Live Upgrade の概要, 234
 - Live Upgrade のためのカスタマイズ, 261
 - Solaris Live Upgrade の概要, 235
 - UFS の作成, 150
 - コピー
 - JumpStart インストールファイル, 147, 151
 - Solaris Live Upgrade の共有可能なファイルシステム, 275
 - マウントされたファイルシステムの表示, 91
 - リモートファイルシステムのマウント, 214
 - ローカルファイルシステムの作成, 215
 - ファクトリ JumpStart, ほかのインストール方法との比較, 30
 - ブートサーバー
 - CD メディアによる作成、例, 87
 - DVD を使用する作成、例, 75
 - サブネット上で DVD メディアを使用して作成, 86
 - サブネット上での作成
 - DVD メディアを使用する, 74
 - 説明, 66
 - ネットワークインストールの要件, 66
 - ブート、システムの、端末とディスプレイを再設定する, 92
 - 複数行のルール、rules ファイル内の, 152
 - フラッシュ
 - flarcreate コマンド, 128
 - flar コマンド
 - アーカイブ情報からの抽出, 131
 - アーカイブの結合, 132
 - アーカイブの分割, 131
 - 説明, 131
 - Live Upgrade ブート環境へのインストール, 286
 - フラッシュインストール機能を参照
 - アーカイブセクションの説明, 125
 - アーカイブを作成する方法, 116
 - インストール
 - インストール方法, 111, 119
 - 階層化されたアーカイブ, 112
 - キーワード
 - identification セクション, 126

キーワード (続き)

- section_begin と section_end, 126
- ほかのインストール方法との比較, 29
- マスターシステム
 - インストール方法, 116
 - 周辺装置, 109
 - 説明, 108
- プラットフォーム
 - インストールサーバーの設定, 90
 - システム属性とプロファイルの一致, 183
 - システム属性とプロファイルの照合, 154
 - システムのモデル名, 200, 229
 - ディスクレスクライアント, 211
 - 名前の決定, 92
 - プローブキーワード, 229
 - ルールキーワード, 199, 229
- プローブキーワード
 - arch, 229
 - disks, 229
 - domainname, 229
 - hostaddress, 229
 - hostname, 229
 - installed, 229
 - karch, 229
 - memsize, 229
 - model, 229
 - network, 229
 - osname, 230
 - totaldisk, 230
- プローブキーワードs, rootdisk, 230
- プロセッサ
 - プローブキーワード, 229
 - ルールキーワード, 197, 229
- フロッピーディスク
 - JumpStart ディレクトリのアクセス, 148
 - フォーマット, 150
- フロッピーディスクのフォーマット, 150
- プロファイル
 - 検査, 160
 - コメント, 156
 - 作成, 155
 - 試験, 61
 - システムの照合, 154, 183
 - 説明, 155
 - テスト, 162
 - 動的プロファイル, 166, 167
 - 要件, 152, 156
 - ルールフィールド, 153

プロファイル (続き)

- 例, 157, 193, 194
- プロファイルキーワード
 - backup_media, 209
 - boot_device, 210
 - client_arch, 211
 - client_root, 212
 - cluster
 - 説明と値, 213
 - 例, 157
 - dontuse
 - usedisk と, 226
 - 説明と値, 214
 - filesystem
 - 説明と値, 214
 - リモートファイルシステム, 214
 - 例, 157
 - ローカルファイルシステム, 215
 - geo
 - 説明と値, 217
 - install_type, 156
 - 説明と値, 219
 - 要件, 157
 - 例, 157
 - isa_bits
 - 説明と値, 219
 - layout_constraint
 - 説明と値, 219
 - locale
 - 説明と値, 221
 - num_clients, 222
 - package
 - 説明と値, 222
 - partitioning
 - 使用しないディスク, 214
 - 説明と値, 223
 - ディスクの指定, 226
 - 例, 157
 - root_device, 224
 - system_type
 - 説明と値, 225
 - 例, 157
 - usedisk
 - 説明と値, 226
- 大文字と小文字の区別, 202
- クラスター
 - 説明と値, 214

プロファイルキーワード>プロファイルキー
ワード, `client_swap`, 212
プロファイルフィールドの = (等号記号), 166
プロファイルフィールドの等号記号 (=), 166

へ

変更、ディレクトリの
JumpStart ディレクトリへ, 163, 181
ローカルディスク上の Solaris 9 ソフトウェ
アのイメージ, 151
ローカルディスク上の Solaris 9 ソフトウェ
アのイメージに, 147

変数

`SI_CONFIG_DIR`, 167
`SI_PROFILE`, 166
`SYS_MEMSIZE`, 161

編成

Solaris 9 DVD, 329
Solaris 9 INSTALLATION CD, 330
Solaris 9 LANGUAGES CD, 332
Solaris 9 SOFTWARE CD, 330

ほ

ホスト名の事前設定, 48
ボリュームマネージャ, コピー, 150

ま

マイクロプロセッサ
プローブキーワード, 229
ルールキーワード, 197, 229
マウント
begin スクリプトの注意事項, 165
Solaris 9 インストールによる, 167
マウントされたファイルシステムの表示,
91
リモートファイルシステム, 214

め

メモリー
スワップ空間のサイズと, 212

メモリー (続き)

プローブキーワード, 229
ルールキーワード, 200, 229

も

問題の解決

一般的なインストールの問題, 349
間違ったサーバーからのブート, 349

よ

要件

`custom_probes` ファイル, 178
Live Upgrade の使用, 247
Solaris Web Start の使用, 25
ディスク容量, 33
ネットワークインストール
サーバー, 65
プロファイル, 152, 156
メモリー, 25

り

リモートファイルシステム, マウント, 214

る

ルート (/) ファイルシステム
JumpStart で設定される値, 224
代替ルート (/) に対するパッケージ要件,
362
ルート (/) ファイルシステム, プロファイルの例
, 142
ルート環境を `finish` スクリプトでカスタマイズ
する, 169
ルール
`rootdisk` 照合ルール, 224
構文, 153
照合順序, 154, 183
妥当性の検査, 164, 182
動的プロファイル, 166, 167
フィールドの説明, 153
複数行のルール, 152

ルール (続き)

例, 154

ルールキーワード

any, 197, 229

arch, 197, 229

disksize, 198, 229

domainname, 198, 229

hostaddress, 198, 229

hostname, 197, 199, 229

installed, 199, 229

karch, 199, 229

memsize, 200, 229

model, 200, 229

network, 200, 229

osname, 201, 230

probe, 201

totaldisk, 202, 230

ろ

ログファイル

begin スクリプト出力, 165

finish スクリプト出力, 167

アップグレードインストール, 98, 104

ロケールファイル, 55

論理積ルールフィールド, 153