



Solaris のシステム管理 (上級編)

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

Part No: 817-0160-10
2002 年 12 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、SunOS、Solstice、Solstice AdminSuite、Solstice DiskSuite、Solaris Solve、Java、JavaStation、DeskSet、OpenWindows、NFS 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。

サンのロゴマークおよび 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、DecWriter、LaserWriter、Epson、NEC、Adobe 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ず、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: *System Administration Guide: Advanced Administration*

Part No: 816-4553-10

Revision A



021120@4879



目次

はじめに 19

1 印刷サービスの管理 (トピック) 25

2 印刷サービスの管理 (概要) 27

印刷の新機能 27

LP スケジューラの変更(lpsched) 27

SPARC: USB プリンタのサポート 28

LDAP によるプリンタ情報の管理 28

プリンタ関連作業の情報の参照箇所 29

Solaris オペレーティング環境での印刷 29

Solaris プリンタマネージャ 30

ネームサービススイッチでの印刷サポート 30

プリンタの管理方法の選択 31

LP 印刷サービス 33

ネットワークプリンタの管理 33

プリンタの管理 33

プリンタの定義の設定 33

文字セット、フィルタ、フォーム、およびフォントの管理 34

LP 印刷サービスのカスタマイズ 34

Solaris 印刷クライアントサーバーの処理手順 34

印刷クライアントの処理手順 34

印刷クライアントの使用 35

プリンタ構成資源 36

プリンタサーバーの使用 38

3	ネットワーク上のプリンタの計画方法 (概要)	41
	プリンタ関連作業の情報の参照箇所	41
	ネットワーク上でのプリンタの分散	42
	プリンタサーバーと印刷クライアントを割り当てる	42
	プリンタサーバーの要件と推奨	43
	スプーリング空間	44
	ディスク空間	44
	メモリー	44
	スワップ空間	45
	ハードディスク	45
	プリンタ設定の計画	45
	プリンタの定義の設定	45
	プリンタタイプの選択	49
	ファイル内容形式の選択	50
4	プリンタの設定手順 (手順)	53
	印刷の設定 (作業マップ)	53
	Solaris プリンタマネージャによる印刷の設定	54
	Solaris プリンタマネージャによるプリンタの設定	55
	Solaris プリンタマネージャと特殊文字	56
	Solaris プリンタマネージャの起動	56
	▼ Solaris プリンタマネージャを起動する方法	56
	プリンタサーバーの設定	59
	▼ 新しく接続するプリンタを Solaris プリンタマネージャを使用して追加する方 法	59
	印刷クライアントの設定	61
	▼ Solaris プリンタマネージャにプリンタアクセスを追加する方法	61
	.printers ファイルの設定	62
	▼ .printers ファイルを設定する方法	63
	ネットワークプリンタの追加	63
	ネットワークプリンタ用のプリンタベンダー提供のソフトウェア	65
	Sun のネットワークプリンタのサポート	65
	ネットワークプリンタサポートの呼び出し	65
	プロトコルの選択	66
	プリンタノード名の選択	66
	宛先 (またはネットワークプリンタアクセス) 名の選択	66
	タイムアウト値の設定	67
	ネットワークプリンタアクセスの管理	68

	▼ プリンタベンダー提供のツールを使用してネットワークプリンタを追加する方法	68
	▼ LP コマンドを使用してネットワークプリンタを追加する方法	69
	プリンタ構成情報を変換する	71
	プリンタ構成情報を変換する (作業マップ)	72
	既存のプリンタ構成情報の変換	72
	▼ Solaris 2.5.1 システムのプリンタ情報を変換する方法	73
	▼ lpd ベースのプリンタサーバーを実行するシステムのプリンタ情報を変換する方法	73
5	プリンタの管理 (手順)	75
	プリンタと印刷スケジューラの管理	76
	プリンタとプリンタアクセスの削除	76
	▼ プリンタとリモートプリンタへのアクセスを削除する方法	76
	プリンタの状態のチェック	79
	▼ プリンタの状態をチェックする方法	79
	印刷スケジューラの再起動	80
	▼ 印刷スケジューラを停止する方法	80
	▼ 印刷スケジューラを再起動する方法	81
	その他のプリンタ定義の設定とリセット	81
	▼ プリンタ記述を追加する方法	81
	デフォルトプリンタの指定	82
	▼ システムのデフォルトプリンタを設定する方法	83
	バナーページの印刷	83
	▼ バナーページを選択可能にする方法	84
	▼ バナーページを抑制する方法	85
	プリンタクラスの設定	86
	▼ プリンタのクラスを定義する方法	87
	障害の通知の設定	87
	▼ プリンタの障害警告を設定する方法	88
	プリンタの障害回復の設定	89
	▼ プリンタの障害回復を設定する方法	90
	プリンタへのユーザーアクセスを制限する	91
	▼ プリンタへのユーザーアクセスを制限する方法	92
	印刷要求の管理	93
	▼ 印刷要求の状態をチェックする方法	94
	プリンタの印刷の有効化と無効化	95
	▼ プリンタを使用可能または使用不可にする方法	96

印刷要求の受け付けまたは拒否	97
▼ プリンタへの印刷要求を受け付けるまたは拒否する方法	98
印刷要求の取り消し	99
▼ 印刷要求を取り消す方法	99
▼ 特定のユーザーからの印刷要求を取り消す方法	100
印刷要求の移動	101
▼ 印刷要求を別のプリンタに移動する方法	101
印刷要求の優先順位の変更	103
▼ 印刷要求の優先順位を変更する方法	103
6 文字セット、フィルタ、フォーム、フォントの管理 (手順)	105
文字セットの管理	106
選択可能な文字セット	106
プリンタに装着する文字セット	107
印字ホイールの確認	108
印字ホイールまたはカートリッジの装着の警告	108
▼ 印字ホイールとフォントカートリッジを定義する方法	108
▼ 印字ホイールまたはフォントカートリッジを取り外すまたは装着する方法	109
▼ 印字ホイールまたはフォントカートリッジの装着を促す警告を設定する方法	111
▼ 選択可能文字セットの別名を設定する方法	112
印刷フィルタの管理	114
印刷フィルタの作成	114
印刷フィルタの追加、変更、削除、および復元	114
▼ 印刷フィルタを追加する方法	116
▼ 印刷フィルタを削除する方法	116
▼ 印刷フィルタに関する情報を表示する方法	117
フォームの管理	118
フォームの追加、変更、または削除	119
フォームの装着	119
フォームの追跡	119
フォームの装着に関する警告の定義	120
フォームのチェック	120
フォームへのアクセスの制限	120
▼ フォームを追加する方法	120
▼ フォームを削除する方法	121
▼ フォームを取り外し、装着する方法	122

▼ フォームの装着に関する警告を設定する方法	123
▼ フォームに関する情報を表示する方法	125
▼ フォームの現在の状態を表示する方法	125
▼ フォームへのユーザーアクセスを制限する方法	126
▼ フォームへのプリンタアクセスを制限する方法	127
フォントの管理	128
プリンタ常駐フォントの管理	129
ホスト常駐フォントのダウンロード	129
ホスト常駐フォントのインストールと管理	130
▼ ダウンロードされた PostScript フォントをインストールする方法	130
▼ ホスト常駐 PostScript フォントをインストールする方法	131
7 LP 印刷サービスのカスタマイズ (手順)	133
プリンタポート特性の調整	133
▼ プリンタポート特性を調整する方法	134
サポートされていないプリンタの terminfo エントリを追加する	135
▼ サポートされていないプリンタの terminfo エントリを追加する方法	138
プリンタインタフェースプログラムのカスタマイズ	139
標準プリンタインタフェースプログラム	139
stty モードのカスタマイズ	140
終了コード	140
障害メッセージ	141
カスタマイズされたプリンタインタフェースプログラムの使用方法	142
▼ 独自のプリンタインタフェースプログラムを設定する方法	142
新しい印刷フィルタの作成	144
印刷フィルタプログラムの作成	144
印刷フィルタ定義の作成	147
▼ 新しい印刷フィルタを作成する方法	153
新しいプリンタフォームの作成	155
▼ 新しいフォーム定義を作成する方法	157
8 LP 印刷サービス (リファレンス)	159
プリンタ関連作業の情報の参照箇所	159
LP 印刷サービス	160
LP 印刷サービスの構造	160
LP 印刷サービスのコマンド	168
LP 印刷サービスの機能	169

	LP によるファイルの管理とローカル印刷要求のスケジューリングの方法	169
	ネットワーク印刷要求のスケジューリング	170
	印刷ファイルにフィルタを適用する	171
	プリンタインタフェースプログラムの機能	171
	lpsched デーモンによる印刷ジョブ状態の確認	171
	ログファイルの消去	172
	▼ プリンタ要求のログの交換間隔を変更する方法	172
	ローカル印刷の処理スケジュール	172
	リモート印刷の処理スケジュール	175
9	端末とモデムの管理 (トピック)	179
10	端末とモデムの管理 (概要)	181
	端末とモデムの管理に関する新機能	181
	端末、モデム、ポート、サービス	181
	端末	182
	モデム	182
	ポート	182
	サービス	183
	ポートモニター	183
	端末とモデムを管理するツール	184
	シリアルポートツール	184
	サービスアクセス機能 (SAF)	184
11	端末とモデムの設定 (手順)	185
	シリアルポートツールによる端末とモデムの設定	185
	端末の設定	186
	モデムの設定	186
	▼ 端末を設定する方法	188
	▼ モデムを設定する方法	189
	▼ ポートを初期化する方法	190
	端末とモデムの問題を解決する方法	190
12	サービスアクセス機能によるシリアルポートの管理 (手順)	193
	サービスアクセス機能 (SAF) の概要	194
	SAF 全体の管理 (sacadm)	195

サービスアクセスコントローラ (SAC プログラム)	195
SAC の初期化プロセス	195
ポートモニターサービス管理 (pmdm)	196
ttymon ポートモニター	196
ポートの初期化プロセス	197
発着信両用サービス	198
TTY モニターとネットワークリスナーポートモニター	198
TTY ポートモニター (ttymon)	198
ttymon とコンソールポート	199
ttymon 固有の管理コマンド (ttyadm)	200
ネットワークリスナーサービス (listen)	200
listen 固有の管理コマンド (nlsadmin)	200
ttymon ポートモニターの管理	201
▼ ttymon ポートモニターを追加する方法	201
▼ ttymon ポートモニターの状態を表示する方法	202
▼ ttymon ポートモニターを停止する方法	203
▼ ttymon ポートモニターを起動する方法	203
▼ ttymon ポートモニターを無効にする方法	203
▼ ttymon ポートモニターを有効にする方法	203
▼ ttymon ポートモニターを削除する方法	204
ttymon サービスの管理	204
▼ サービスを追加する方法	204
▼ TTY ポートサービスの状態を表示する方法	205
▼ ポートモニターサービスを有効にする方法	208
▼ ポートモニターサービスを無効にする方法	208
サービスアクセス機能の管理 (リファレンス)	208
SAF の関連ファイル	208
/etc/saf/_sactab ファイル	209
/etc/saf/pmtab/_pmtab ファイル	210
サービスの状態	211
ポートモニターの状態	211
ポートの状態	212
13 システム資源の管理 (トピック)	213
14 システム資源の管理 (概要)	215
システム資源管理の新機能	215

- df、du、ls の新オプション 215
 - システムパラメータの変更 216
 - Solaris 資源管理 216
 - システム資源管理の概要 (作業マップ) 217
- 15 システム情報の表示と変更 (手順) 219**
- システム情報の表示 219
 - ▼ システムが 64 ビット Solaris オペレーティング環境を実行できるか調べる方法 220
 - ▼ 64 ビット Solaris 機能が有効になっているか調べる方法 221
 - ▼ Solaris リリース情報を表示する方法 222
 - ▼ 一般的なシステム情報を表示する方法 222
 - ▼ 一般的なシステム情報を表示する方法 222
 - ▼ システムのホスト ID を表示する方法 223
 - ▼ システムにインストールされているメモリーを表示する方法 223
 - ▼ 日付と時刻を表示する方法 223
 - システム情報の変更 224
 - ▼ システムの日付と時刻を手作業で設定する方法 224
 - ▼ その日のメッセージを設定する方法 225
 - ▼ システムのホスト名の変更方法 225
- 16 ディスク使用の管理 (手順) 227**
- ファイルとディスク容量の情報の表示 228
 - ▼ ファイルとディスク容量の情報を表示する方法 228
 - ファイルサイズの確認 230
 - ▼ ファイルサイズを表示する方法 230
 - ▼ サイズの大きなファイルを見つける方法 231
 - ▼ 指定されたサイズ制限を超えるファイルを見つける方法 232
 - ディレクトリサイズの確認 233
 - ▼ ディレクトリ、サブディレクトリ、およびファイルのサイズを表示する方法 233
 - ▼ ローカル UFS ファイルシステムのユーザーの所有状況を表示する方法 234
 - 古いファイルまたは使用されていないファイルの検索と削除 235
 - ▼ 最新ファイルのリストを表示する方法 236
 - ▼ 古いファイルと使用されていないファイルを見つけて削除する方法 236
 - ▼ 一時ディレクトリを一度にクリアする方法 237
 - ▼ core ファイルを見つけて削除する方法 238

▼クラッシュダンプファイルを削除する方法 239

- 17 ディスク割り当て制限の管理 (手順) 241
 - ディスクの割り当て制限 241
 - ディスク割り当て制限の使用 242
 - ディスク割り当て制限の弱い制限値と強い制限値の設定 242
 - ディスクブロックとファイル制限の相違 243
 - ディスク割り当て制限の設定 243
 - 割り当て制限設定のガイドライン 244
 - 割り当て制限の設定 (作業マップ) 245
 - ▼割り当て制限用にファイルシステムを構成する方法 245
 - ▼1 ユーザーに割り当て制限を設定する方法 246
 - ▼複数ユーザーに割り当て制限を設定する方法 247
 - ▼割り当て制限の整合性を確認する方法 248
 - ▼割り当て制限を有効にする方法 249
 - 割り当て制限のチェック 249
 - ▼割り当て制限を超過したかどうかを確認する方法 250
 - ▼ファイルシステムの割り当て制限を確認する方法 250
 - 割り当て制限の変更と削除 252
 - ▼弱い期間制限値のデフォルトを変更する方法 252
 - ▼1 ユーザーの割り当て制限を変更する方法 253
 - ▼1 ユーザーの割り当て制限を無効にする方法 254
 - ▼割り当て制限を無効にする方法 255

- 18 システムタスクのスケジュール設定 (手順) 257
 - システムタスクを自動的に実行する方法 257
 - 繰り返されるジョブのスケジューリング (crontab) 258
 - 1つのジョブのスケジューリング (at) 259
 - 繰り返されるシステムタスクのスケジューリング (cron) 259
 - crontab ファイルの内容 260
 - cron デーモンのスケジューリング 261
 - crontab ファイルエントリの構文 261
 - crontab ファイルの作成と編集 262
 - ▼ crontab ファイルを作成または編集する方法 262
 - ▼ crontab ファイルを確認する方法 263
 - crontab ファイルの表示 264
 - ▼ crontab ファイルを表示する方法 264

	crontab ファイルの削除	265
	▼ crontab ファイルを削除する方法	265
	crontab コマンドの使用制御	266
	▼ crontab の使用を拒否する方法	267
	▼ crontab の使用を特定のユーザーに限定する方法	267
	▼ crontab の使用制限を確認する方法	268
	1 つのシステムタスクのスケジューリング (at)	269
	at コマンドの説明	269
	▼ at ジョブを作成する方法	270
	▼ at 待ち行列を表示する方法	271
	▼ at ジョブを確認する方法	271
	▼ at ジョブを表示する方法	271
	▼ at ジョブを削除する方法	272
	▼ at コマンドの使用を拒否する方法	273
	▼ at の使用の拒否を確認する方法	273
19	システムアカウントिंगの設定と管理 (手順)	275
	システムアカウントिंग	275
	システムアカウントिंगの動作	276
	システムアカウントिंगのコンポーネント	276
	システムアカウントिंगの設定	280
	▼ システムアカウントिंगを設定する方法	281
	ユーザーへの課金	283
	▼ ユーザーに課金する方法	283
	アカウントिंग情報の管理	284
	壊れたファイルと wtmpx エラーを修復する	284
	▼ 壊れた wtmpx ファイルの修復方法	284
	tacct エラーを修復する	284
	▼ tacct エラーを修復する方法	285
	runacct スクリプトを再起動する	285
	▼ runacct スクリプトを再起動する方法	286
	システムアカウントिंगの停止と無効	286
	▼ 一時的にシステムアカウントिंगを停止する方法	286
	▼ システムアカウントिंगを永続的に無効にする方法	287
20	システムアカウントिंग (リファレンス)	289
	runacct スクリプト	289

	日次アカウントングレポート	292
	日次レポート	292
	日次利用状況レポート	293
	日次コマンド要約	295
	月次コマンド要約	296
	最終ログインレポート	297
	acctcom による pacct ファイルの確認	297
	システムアカウントングファイル	299
	runacct スクリプトが生成するファイル	301
21	システムパフォーマンスの管理 (トピック)	303
22	システムパフォーマンスの管理 (概要)	305
	システムパフォーマンスの管理に関する新機能	305
	DNLC 機能の向上	305
	pargs コマンドと preap コマンド	306
	パフォーマンスツール	307
	新しい FSS スケジューラと FX スケジューラ	307
	システムパフォーマンスに関する情報の参照箇所	308
	システムパフォーマンスおよびシステム資源	309
	プロセスとシステムのパフォーマンス	309
	パフォーマンスの監視	311
	監視ツール	312
23	システムプロセスの管理 (手順)	313
	システムプロセスの管理に使用するコマンド (概要)	313
	ps コマンド	314
	▼ プロセスを表示する方法	315
	/proc ファイルシステムとコマンド	316
	/proc プロセスコマンドを使用したプロセスの管理	317
	▼ プロセスに関する情報を表示する方法	318
	▼ プロセスを制御する方法	319
	プロセスの終了 (pkill)	320
	▼ プロセスを終了させる方法	320
	プロセスクラス情報の管理	321
	プロセスのスケジュール優先順位を変更する (priocntl)	321
	▼ プロセスクラスに関する基本情報を表示する方法 (priocntl)	321

- ▼ プロセスのグローバル優先順位を表示する方法 (ps) 322
 - ▼ プロセスの優先順位を指定する方法 (priocntl) 323
 - ▼ タイムシェアリングプロセスのスケジューリングパラメータを変更する方法 (priocntl) 323
 - ▼ プロセスのクラスを変更する方法 (priocntl) 324
 - タイムシェアリングプロセスの優先順位を変更する (nice) 325
 - ▼ プロセスの優先順位を変更する方法 (nice) 326
 - システムのプロセスに関する問題解決方法 327
- 24 システムパフォーマンスの監視 (手順) 329**
- システムパフォーマンスの監視に関する新機能 330
 - 仮想メモリーの統計情報の表示 (vmstat) 330
 - ▼ 仮想メモリーの統計情報を表示する方法 (vmstat) 331
 - ▼ システムイベント情報を表示する方法 (vmstat -s) 332
 - ▼ スワップの統計情報を表示する方法 (vmstat -S) 333
 - ▼ キャッシュフラッシュの統計情報を表示する方法 (vmstat -c) 333
 - ▼ 各デバイス当りの割り込み数を表示する方法 (vmstat -i) 334
 - ディスク使用状況の表示 (iostat n) 334
 - ▼ ディスクの使用状況を表示する方法 (iostat) 335
 - ▼ 拡張ディスク統計情報を表示する方法 (iostat -xtc) 336
 - ディスク容量統計情報の表示 (df) 337
 - ▼ ディスク容量情報を表示する方法 (df) 337
 - システム動作の監視 (sar) 338
 - ▼ ファイルアクセスをチェックする方法 (sar -a) 338
 - ▼ バッファ動作をチェックする方法 (sar -b) 339
 - ▼ システムコールの統計情報をチェックする方法 (sar -c) 340
 - ▼ ディスク動作をチェックする方法 (sar -d) 341
 - ▼ ページアウトとメモリーをチェックする方法 (sar -g) 343
 - ▼ カーネルメモリーの割り当てをチェックする方法 (sar -k) 344
 - ▼ プロセス間通信をチェックする方法 (sar -m) 345
 - ▼ ページイン動作をチェックする方法 (sar -p) 346
 - ▼ 待ち行列動作をチェックする方法 (sar -q) 347
 - ▼ 未使用のメモリーをチェックする方法 (sar -r) 348
 - ▼ CPU の使用状況をチェックする方法 (sar -u) 349
 - ▼ システムテーブルの状態をチェックする方法 (sar -v) 351
 - ▼ スワップ動作をチェックする方法 (sar -w) 352
 - ▼ 端末動作をチェックする方法 (sar -y) 353
 - ▼ システム全体のパフォーマンスをチェックする方法 (sar -A) 354

	システム動作データの自動収集 (sar)	354
	ブート時に sadc コマンドを実行する	355
	sa1 スクリプトを使用して sadc コマンドを定期的に行う	355
	sa2 スクリプトを使用してレポートを生成する	355
	データの自動収集を設定する (sar)	355
	自動データ収集を設定する方法	357
25	Solaris ソフトウェアで発生する問題の解決 (トピック)	359
26	ソフトウェアの問題解決 (概要)	361
	ソフトウェアの問題解決に関する新機能	361
	新しいシステムログローテーション	361
	新しい root アカウントに対するシェルのフォールバック	362
	ソフトウェアの問題の解決方法の参照先	363
	システムクラッシュの問題の解決	363
	システムがクラッシュした場合の対処方法	363
	問題の解決に使用するデータの収集	364
	システムクラッシュを解決するためのチェックリスト	365
	システムメッセージの表示	366
	▼ システムメッセージを表示する方法	367
	システムメッセージ記録のカスタマイズ	368
	▼ システムのメッセージ記録をカスタマイズする方法	369
	リモートコンソールメッセージングを有効にする	370
	実行レベルの変更中に補助コンソールメッセージングを使用する	371
	対話型ログインセッション中に consadm コマンドを使用する	372
	▼ 補助 (リモート) コンソールを有効にする方法	372
	▼ 補助コンソールのリストを表示する方法	373
	▼ システムリブート後も補助 (リモート) コンソールを有効にする方法	373
	▼ 補助 (リモート) コンソールを無効にする方法	374
27	コアファイルの管理 (手順)	375
	コアファイルの管理 (作業マップ)	375
	コアファイルの管理の概要	376
	構成可能なコアファイルのパス	376
	拡張されたコアファイル名	377
	コアファイル名パターンの設定	377
	setuid プログラムがコアファイルを作成できるようにする	378

- 現在のコアダンプ構成を表示する方法 378
- コアファイル名パターンを設定する方法 379
 - ▼ プロセス別コアファイルパスを有効にする方法 379
 - ▼ グローバルのコアファイルパスを有効にする方法 379
- コアファイルの問題解決 380
- コアファイルの検査 380
 - 例 — proc ツールを使用したコアファイルの検査 380

- 28 システムクラッシュ情報の管理 (手順) 381**
 - システムクラッシュ情報 (作業マップ) 381
 - システムクラッシュ (概要) 382
 - システムクラッシュダンプファイル 382
 - クラッシュダンプの保存 383
 - dumpadm コマンド 383
 - ダンプデバイスとボリュームマネージャ 384
 - システムクラッシュダンプ情報の管理 385
 - ▼ 現在のクラッシュダンプ構成を表示する方法 385
 - ▼ クラッシュダンプ構成を変更する方法 385
 - ▼ クラッシュダンプを検査する方法 387
 - ▼ クラッシュダンプディレクトリが一杯になった場合に復元する方法 (省略可能) 387
 - ▼ クラッシュダンプの保存を無効または有効にする方法 388

- 29 ソフトウェアで発生するさまざまな問題の解決 (手順) 389**
 - リポートが失敗した場合の対処 389
 - SPARC: 64 ビット Solaris のブートで発生する問題の解決 390
 - ルートパスワードを忘れた場合の対処 391
 - SPARC: 例 — ルートパスワードを忘れた場合の対処方法 391
 - IA: 例 — ルートパスワードを忘れた場合の対処方法 392
 - システムがハングした場合の対処 393
 - ファイルシステムが一杯になった場合の対処 394
 - 大規模ファイルまたはディレクトリを作成したために、ファイルシステムが一杯になる 395
 - システムのメモリーが不足したために、tmpfs ファイルシステムが一杯になる 395
 - コピーまたは復元後にファイルの ACL が消失した場合の対処 395
 - バックアップ時の問題の解決 396

	ファイルシステムのバックアップ中に、ルート (/) ファイルシステムが一杯になる	396
	バックアップコマンドと復元コマンドが対応していることを確認する	396
	現在のディレクトリが間違っていないことを確認する	397
30	ファイルアクセスでの問題の解決 (手順)	399
	検索パスに関連する問題を解決する (コマンドが見つかりません)	399
	▼ 検索パスの問題を診断して解決する方法	400
	ファイルアクセスの問題を解決する	402
	ファイルとグループの所有権の変更	402
	ネットワークアクセスで発生する問題の把握	402
31	印刷時の問題の解決 (手順)	405
	印刷時の問題解決のヒント	405
	出力されない (印刷されない) 場合の対処	406
	出力が正しくない場合の対処	407
	LP コマンドがハングした場合の対処	409
	プリンタがアイドル状態になった (ハングした) 場合の対処	409
	矛盾した状態メッセージの対処	410
	印刷時の問題の解決	411
	▼ プリンタに出力されない問題を解決する方法	411
	▼ プリンタハードウェアをチェックする方法	411
	▼ プリンタのネットワーク接続をチェックする方法	412
	▼ LP 印刷サービスの基本機能をチェックする方法	413
	▼ Solaris 印刷クライアントから Solaris プリンタサーバーへの印刷をチェックする方法	416
	▼ 出力が正しくない場合の問題を解決する方法	417
	▼ LP 印刷サービスのハングを解除する方法	421
	▼ アイドル状態になった (ハングした) プリンタの問題を解決する方法	422
	▼ プリンタの準備ができていないかチェックする方法	422
	▼ 印刷のフィルタリングをチェックする方法	422
	▼ プリンタ障害の後に印刷を再開する方法	423
	▼ ローカル待ち行列で停止している印刷要求をリモートプリンタに送信する方法	423
	▼ プリンタサーバーの待ち行列で停止する印刷クライアントからの印刷要求を使用可能にする方法	423
	▼ 矛盾したプリンタ状態メッセージを解決する方法	424

32	UFS ファイルシステムの不整合解決 (手順)	425
	fsck エラーメッセージ	425
	fsck の一般エラーメッセージ	427
	初期化フェーズでの fsck メッセージ	428
	フェーズ 1: ブロックとサイズの検査のメッセージ	431
	フェーズ 1B: 走査し直して DUPS メッセージを表示する	434
	フェーズ 2: パス名の検査のメッセージ	434
	フェーズ 3: 接続性の検査のメッセージ	440
	フェーズ 4: 参照数の検査のメッセージ	442
	フェーズ 5: シリンダグループ検査のメッセージ	445
	クリーンアップ (後処理) フェーズのメッセージ	446
33	ソフトウェアパッケージで発生する問題の解決 (手順)	449
	ソフトウェアパッケージのシンボリックリンクに関する問題の解決	449
	特定のソフトウェアパッケージのインストールエラー	450
	一般的なソフトウェアパッケージのインストールに関する問題	451
	索引	453

はじめに

本書『Solaris のシステム管理 (上級編)』は、Solaris™ システム管理に関する重要な情報を提供するマニュアルの一部です。SPARC™ プラットフォームおよび IA プラットフォームにおけるシステム管理について説明します。

このマニュアルでは、システム管理者である読者が SunOS™ 5.9 オペレーティングシステムをすでにインストールして、ネットワークソフトウェアの設定を終了していることを想定しています。SunOS 5.9 オペレーティングシステムは Solaris 9 製品の一部で、Solaris 9 には Solaris 共通デスクトップ環境 (CDE) などの多くの機能が含まれています。また、SunOS 5.9 は、AT&T System V リリース 4 オペレーティングシステムに準拠しています。

システム管理者にとって重要な Solaris 9 リリースの新機能については、各章のはじめにある新機能に関する節を参照してください。

注 – Solaris オペレーティング環境は、SPARC と IA という 2 種類のハードウェア、つまりプラットフォームで動作します。また、Solaris オペレーティング環境は、64 ビットのアドレス空間でも 32 ビットのアドレス空間でも動作します。このマニュアルで説明する情報は、章、節、注、箇条書き、図、表、例、またはコード例において特に明記しないかぎり、両方のプラットフォームおよびアドレス空間に該当します。

対象読者

このマニュアルは、Solaris 9 リリースを実行するシステムの管理者を対象にしています。このマニュアルを読むには、UNIX のシステム管理について 1～2 年の経験が必要です。UNIX システム管理のトレーニングコースに参加することも役に立ちます。

『Solaris のシステム管理』の構成

システム管理マニュアルで説明している内容を次に示します。

マニュアル名	内容
『Solaris のシステム管理 (基本編)』	ユーザーアカウントとグループ、サーバーとクライアントのサポート、システムのシャットダウンとブート、リムーバブルメディア、ソフトウェアの管理 (パッケージとパッチ)、ディスクとデバイス、ファイルシステム、およびデータのバックアップと復元
『Solaris のシステム管理 (上級編)』	印刷サービスの管理、端末とモデムの設定、システム資源の管理 (ディスク割り当て、アカウントティング、および <code>crontab</code> ファイルの管理)、システムプロセスの管理、および Solaris ソフトウェアの障害追跡
『Solaris のシステム管理 (IP サービス)』	TCP/IP ネットワークの管理、IPv4 から IPv6 への移行と IPv6 の管理、DHCP の管理、IP セキュリティの管理、モバイル IP の管理、および IP ネットワークマルチパスの管理
『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)』	DNS、NIS、および LDAP のネーミングとディレクトリサービスの管理
『Solaris のシステム管理 (ネーミングとディレクトリサービス : FNS、NIS+ 編)』	FNS および NIS+ のネーミングとディレクトリサービスの管理
『Solaris のシステム管理 (資源管理とネットワークサービス)』	システム資源、リモートファイルシステム、メールサービス、SLP、および PPP の管理
『Solaris のシステム管理 (セキュリティサービス)』	監査、PAM、RBAC、および SEAM の管理

Solaris オペレーティング環境の今回のリリースに含まれるオープンソースソフトウェアのライセンス規約、帰属、著作権の記載内容を見るためのデフォルトパスは、`/usr/share/src/freeware-name` または `/usr/sfw/share/src/freeware-name` です。Solaris オペレーティング環境がデフォルト以外の場所にインストールされている場合は、所定のパスを修正して、インストールした場所にあるファイルにアクセスします。

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せん。

表 P-1 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。	<code>.login</code> ファイルを編集します。 <code>ls -a</code> を使用してすべてのファイルを表示します。 <code>system%</code>
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。	<code>system% su</code> <code>password:</code>
<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、 <code>rm filename</code> と入力します。
『』	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。	<code>sun% grep '^#define \</code> <code>XV_VERSION_STRING'</code>

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 III Xeon、Celeron、Pentium III、Pentium III Xeon、Pentium 4 の各プロセッサ、および AMD 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。

- このマニュアル中の手順を実行したり、例 (コマンド入力、コードなど) を使用する場合には、二重引用符 (""), 左一重引用符 ('), 右一重引用符 (') をそれぞれ間違えないように注意してください。
- このマニュアル中で「Return キー」と表記しているキーは、キーボードによっては「Enter キー」という名前になっていることがあります。
- /sbin、/usr/sbin、/usr/bin、/etc ディレクトリにあるコマンドについては、このマニュアルでは絶対パス名で表記していない場合があります。

- このマニュアル中の例は、SunOS 5.9 ソフトウェアの基本部分がインストールされ、バイナリ互換パッケージがインストールされておらず、`/usr/ucb` が検索パスに設定されていないことを、前提としています。



注意 - `/usr/ucb` を検索パスに設定する場合は、パスの一番最後に設定してください。ps コマンドや df コマンドなどは、SunOS 5.9 コマンドと `/usr/ucb` コマンドとで形式やオプションがそれぞれ異なります。

第 1 章

印刷サービスの管理 (トピック)

以下の各章で、印刷サービスの管理について説明します。

第 2 章	ネットワーク上で印刷サービスを管理する場合の概要について説明します。プリンタサーバー、印刷クライアント、LP 印刷サービスを取り上げます。
第 3 章	ネットワーク上で印刷サービスの計画を作成する場合の概要について説明します。ネットワーク上のシステム資源の割り当てやプリンタの定義が含まれます。
第 4 章	システム上でプリンタを設定し、ネットワーク上の他のシステムが利用できるようにする詳しい手順を説明します。
第 5 章	プリンタの削除、印刷ポリシーの設定、印刷要求の管理など、プリンタを管理する詳しい手順を説明します。
第 6 章	文字セット、印刷フィルタ、フォーム、およびフォントを設定し管理する詳しい手順を説明します。
第 7 章	プリンタポート特性の調整や、サポートされていないプリンタの terminfo エントリの追加など、LP 印刷サービスをカスタマイズする詳しい手順を説明します。
第 8 章	LP 印刷サービスの参照情報を説明します。

第 2 章

印刷サービスの管理 (概要)

この章では、Solaris 環境のプリンタの管理について基本的なことから説明します。

この章の内容は以下のとおりです。

- 27 ページの「印刷の新機能」
- 29 ページの「Solaris オペレーティング環境での印刷」
- 33 ページの「LP 印刷サービス」
- 34 ページの「Solaris 印刷クライアントサーバーの処理手順」

印刷の新機能

この節では、Solaris 9 リースの新しい印刷機能について説明します。

LP スケジューラの変更(lpsched)

これまでの Solaris リリースでは、システムにローカルプリンタが設定されているかどうかに関わらず、システムがブートすると、自動的に lpsched プロセスが起動しました。

今回の Solaris リリースでは、ローカルプリンタがシステムに追加されると、lpadmin コマンドが自動的に lpsched プロセスを起動し、ローカルプリンタが削除されると、プロセスを停止します。

この変更によって、印刷クライアントの側から、印刷サブシステムの使用方法や管理方法が変わることはありません。lpsched プロセスは、ローカルプリンタが設定されていないシステムでは実行しません。このことが、印刷や新しいプリンタの設定に影響することはありません。

SPARC: USB プリンタのサポート

Solaris プリンタマネージャを使用すると、SPARC システムまたは IA システムに USB ポートで接続した USB プリンタをセットアップすることができます。

USB プリンタ用の新しい論理デバイス名は次のとおりです。

```
/dev/printers/[0...N]*
```

プリンタサーバーに USB プリンタを追加する場合は、「新しいローカルプリンタを設定 (Add New Attached Printer)」の「プリンタポート (Printer Port)」画面から、これらの USB プリンタ用デバイスのいずれかを選択します。Solaris プリンタマネージャを使用してプリンタを設定する方法については、第 4 章を参照してください。

新しい Solaris USB プリンタドライバは USB プリンタクラス準拠のプリンタをすべてサポートします。推奨される PostScript™ プリンタについては、usbprn (7D) のマニュアルページを参照してください。

usbprn ドライバは、GhostScript など Sun 社製以外の PostScript 変換パッケージを使用する PostScript 以外のプリンタでも使用できます。変換パッケージは Solaris Software Companion CD から入手できます。Solaris Software Companion CD の入手方法については、<http://www.sun.com/software/solaris/binaries/package.html> を参照してください。

USB プリンタのホットプラグインの詳細と注意事項については、usbprn (7D) のマニュアルページの「Notes」と「Diagnostics」の節を参照してください。

LDAP によるプリンタ情報の管理

今回の Solaris リリースでは、次のネームサービスでプリンタ情報を管理することができます。

- LDAP
- NIS
- NIS+
- FNS を使用する NIS+
- ファイル

LDAP ネームサービスでプリンタ情報を管理する場合は、次の事項に留意してください。

- LDAP サーバが Netscape™ ディレクトリサーバー (NSDS) の場合、デフォルトの識別名は `cn=Directory Manager` となります。LDAP サーバが Sun™ ディレクトリサーバーの場合、識別名は `cn=admin, dc=xyz, dc=com` のようになります。Solaris プリンタマネージャは、`ldapclient` コマンドを使用してデフォルトの LDAP サーバを決めます。複数のサーバが指定されている場合は、最初のサーバが自動的に選択されます。詳細は、`ldapclient (1M)` のマニュアルページを参照してください。

- Solaris プリンタマネージャは、常にプリンタエントリを現在の LDAP サーバーから表示します。ドメインの LDAP マスターサーバーでない場合は、表示されるプリンタのリストは、プリンタの現在のリストではないかもしれません。LDAP 複製サーバーはマスターサーバーによって更新されていないため、マスターと同期していないことがあるからです。複製サーバーは、さまざまな更新複製規約を持つことができます。たとえば、マスターに変更があった場合に、複製サーバーがただちに更新されるようにしたり、一日に一回更新されるようにしたりすることができます。
- 選択した LDAP サーバーが LDAP 複製サーバーである場合、あらゆる更新はマスターサーバーを参照し、そこで行われます。この状態は、プリンタのリストがマスターと同期していないことを意味します。たとえば、複製がマスターから更新されるまで、削除したプリンタがプリンタリストに表示されたままになります。
- ユーザーは、LDAP コマンド行ユーティリティ、`ldapadd` および `ldapmodify` を使用して、ディレクトリ内のプリンタエントリを更新できますが、これはお奨めしません。これらのユーティリティを使用する場合、ユーザーは `printer-name` 属性値が `ou=printers` コンテナ内で一意であることを確かめる必要があります。`printer-name` 属性値が一意でない場合、Solaris プリンタマネージャや `lpset` コマンドによる修正の結果は、予測できないものとなります。

プリンタ関連作業の情報の参照箇所

プリンタ関連作業	参照箇所
Solaris プリンタマネージャと <code>lp</code> コマンドによるプリンタのセットアップ	第 4 章
セットアップ後の <code>lp</code> コマンドによるプリンタの管理	第 5 章
文字セット、フィルタ、フォーム、フォントの管理	第 6 章
LP 印刷サービスのカスタマイズ	第 7 章

Solaris オペレーティング環境での印刷

Solaris 印刷ソフトウェアは、ネットワーク上のプリンタへのクライアントアクセスを設定および管理するための環境を提供します。

Solaris 印刷ソフトウェアは、次のコンポーネントから構成されます。

- Solaris プリンタマネージャ。ローカルシステム上やネームサービス内の印刷構成を管理するグラフィカルユーザーインターフェースです。
- LP 印刷サービスコマンド。ローカルシステム上やネームサービス内のプリンタを設定して管理するコマンド行インターフェースです。上記の機能に加え、他の印刷管理ツールにない機能も提供します。

印刷の設定には Solaris プリンタマネージャを使用しますが、Solaris 環境で印刷を完全に制御するためには、LP コマンドの一部を使用する必要があります。詳細は、第 5 章を参照してください。

Solaris 印刷ソフトウェアの制限事項には次のものがあります。

- 以前の Solaris リリースで s5 (System V 印刷プロトコル) として定義されたプリンタサーバーはサポートされません。
- 印刷クライアントでの印刷のフィルタリングはできません。

Solaris プリンタマネージャ

Solaris プリンタマネージャは Java ベースのグラフィカルユーザーインターフェースで、ローカルおよびリモートのプリンタ構成を管理できます。このツールは、ネームサービス環境 (NIS、NIS+、フェデレーテッド・ネーミング・サービス (FNS) を使用した NIS+、およびファイル) 内で使用できます。このツールを使用するには、スーパーユーザーとしてログインする必要があります。

プリンタ構成情報の管理には、Admintool: プリンタ (Printer) よりも Solaris プリンタマネージャの使用をお勧めします。Solaris プリンタマネージャをネームサービスとともに使用すれば、プリンタ情報を一元的に管理できるためです。プリンタ構成情報の格納には、ネームサービスの使用をお勧めします。ネームサービスを使用すると、ネットワーク上のすべてのシステムからプリンタ情報にアクセスできるようになるためです。これにより、印刷管理がより簡単になります。

Solaris プリンタマネージャは、プリンタサーバー、印刷クライアント、ネームサービスデータベースにあるプリンタ情報を認識します。印刷クライアントで Solaris 2.6、Solaris 7、Solaris 8 または Solaris 9 リリースが動作している限り、新しい Solaris プリンタマネージャを使用するために変換作業は必要ありません。

Solaris プリンタマネージャのパッケージは SUNWppm です。

Solaris プリンタマネージャを使用して新しいプリンタを設定する詳しい手順を知りたい場合は、第 4 章に進みます。

ネームサービススイッチでの印刷サポート

ネームサービススイッチファイル /etc/nsswitch.conf 内の printers データベースは、ネットワーク上のクライアントを出力するために、一元化したプリンタ構成情報を提供します。

ネームサービススイッチファイルに `printers` データベースとそれに対応する情報源を指定すると、印刷クライアントからプリンタ構成情報に自動的にアクセスできるようになるため、この情報を自分のシステムに追加する必要はありません。

ファイル、LDAP、NIS および NIS+ 環境に対応した `/etc/nsswitch.conf` ファイル内のデフォルトの `printers` エントリについて、次の表で説明します。 `nisplus` キーワードは `printers.org_dir` テーブルを表します。キーワード `xfn` は、FNS プリンタコンテキストを示します。

ネームサービスタイプ	デフォルトの <code>printers</code> エントリ
<code>files</code>	<code>printers: user files</code>
<code>ldap</code>	<code>printers: user files ldap</code>
<code>nis</code>	<code>printers: user files nis</code>
<code>nis+</code>	<code>printers: user nisplus files xfn</code>

たとえば、ネームサービスが NIS の場合、印刷クライアントのプリンタ構成情報は次のソースから次に記載する順に検索されます。

- `user` - ユーザーの `$HOME/.printers` ファイルを表します。
- `files` - `/etc/printers.conf` ファイルを表します。
- `nis` - `printers.conf.byname` テーブルを表します。

詳細は、`nsswitch.conf` (4) のマニュアルページおよび『*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。

プリンタの管理方法の選択

プリンタ情報をネームサービスに追加すると、ネットワークのすべてのシステムからプリンタにアクセスできるようになります。さらに、プリンタに関するすべての情報が一元化されるため、プリンタ管理が一般に簡単になります。

ネームサービス構成	プリンタ情報を一元化する方法
ネームサービスを使用する	プリンタを LDAP、NIS、NIS+、NIS+ (<code>xfn</code>) データベースのどれかに追加すると、プリンタはネットワーク上のすべてのシステムからアクセスできるようになる

ネームサービス構成	プリンタ情報を一元化する方法
ネームサービスを使用しない	<p>プリンタを追加しても、プリンタ情報はプリンタサーバーの構成ファイルにしか追加されない。したがって、印刷クライアントがそのプリンタを自動的に認識することはできない。</p> <p>プリンタを必要とする印刷クライアントにはプリンタ情報を追加する必要がある</p>

表 2-1 に、印刷関連の主な作業と、印刷作業に利用できるツールを示します。

表 2-1 Solaris 印刷コンポーネントの機能

コンポーネント	対応するリリース	グラフィカル ユーザーインタ フェース	ネットワークプ リンタの設定	印刷クライアント とプリンタ サーバーの管理	LDAP、NIS、 NIS+、NIS+ (xfn) の使用
Solaris プリン タマネージャ	Solaris 9 およ び互換パー ジョンおよび Solaris Easy Access Server 3.0	あり	できる	できる	できる
Admintool	Solaris 9 およ び互換パー ジョン	あり	できない	できる	できない
LP コマンド	Solaris 9 およ び互換パー ジョン	なし	できる	できる	できる

表 2-1 を使用して、各ネットワーク環境に最適な印刷ツールを決めてから、プリンタの設定情報について第 4 章を参照してください。

印刷構成作業のほとんどは Solaris プリンタマネージャで完了設定できます。ただし、インタフェーススクリプトの作成や独自フィルタの追加など、特別な必要がある場合は、LP 印刷サービスコマンドを使用します。LP コマンドは、Solaris プリンタマネージャの元となるコマンドです。LP コマンドで印刷管理作業をする方法については、第 5 章で説明しています。

LP 印刷サービス

「LP 印刷サービス」とは、ユーザーが作業を続けながらファイルを印刷できるようにするソフトウェアユーティリティの集合です。

LP 印刷サービスの基本情報については、第 8 章を参照してください。

その他に取り上げられている LP 印刷サービスの事項と参照する章については、次の節で説明します。

ネットワークプリンタの管理

「ネットワークプリンタ」はネットワークに直接接続されているハードウェアデバイスであり、ネットワークを介して出力デバイスにデータを直接送信します。プリンタやネットワークに接続されたハードウェアには、固有のシステム名と IP アドレスがあります。

一般に、ネットワークプリンタには、プリンタベンダーから提供されているソフトウェアが必要です。プリンタにベンダー支給のソフトウェアがある場合は、プリンタベンダーのソフトウェアを使用します。ネットワークプリンタのベンダーがソフトウェアサポートを提供していない場合には、Sun が提供するソフトウェアを利用できます。このソフトウェアはネットワークプリンタの汎用サポートを提供するもので、必ずしもプリンタが持つすべての機能を利用できるわけではありません。

ネットワークプリンタの設定手順については、第 4 章を参照してください。

プリンタの管理

プリンタサーバーと印刷クライアントを設定すると、次の管理作業を実行する必要性が頻繁に生じる可能性があります。

- プリンタとリモートプリンタへのアクセスを削除する
- プリンタの状態をチェックする
- 印刷スケジューラを再起動する

プリンタ管理作業を実行する手順については、第 5 章を参照してください。

プリンタの定義の設定

ネットワーク上でのプリンタの定義は、より効率的な印刷環境をユーザーに提供するための継続的な作業です。たとえば、サイトにあるすべてのプリンタに説明を付ければ、ユーザーはプリンタがどこにあるのかを見つけやすくなります。あるいは、プリンタのクラスを定義することにより、印刷要求を迅速に処理できます。

プリンタ定義の設定については、第3章を参照してください。

文字セット、フィルタ、フォーム、およびフォントの管理

設置要件とネットワーク上のプリンタの種類に応じて、LP 印刷サービスのプリンタ固有の機能を設定して管理する必要があります。たとえば、異なる印字ホイール、フィルタ、フォームをプリンタごとに割り当てることができます。文字セット、印刷フィルタ、フォーム、フォントの設定方法に関する基本情報と手順は、第6章を参照してください。

LP 印刷サービスのカスタマイズ

LP 印刷サービスは、ほとんどのプリンタと印刷ニーズに対応できる十分な柔軟性を持つように設計されていますが、あらゆる状況に対処できるわけではありません。LP 印刷サービスの標準機能では対処できない印刷要求が発生する場合があります。また、LP 印刷サービスによるプリンタの処理方法に当てはまらないプリンタを使用することもあります。

LP 印刷サービスは、次のような方法でカスタマイズできます。

- プリンタポートの特性を調整する
- terminfo データベースを調整する
- プリンタインタフェースプログラムをカスタマイズする
- 印刷フィルタを作成する
- フォームを定義する

LP 印刷サービスのカスタマイズの詳細と手順については、第7章を参照してください。

Solaris 印刷クライアントサーバーの処理手順

この節では、Solaris で印刷がどのように進むかについて概要を説明します。

印刷クライアントの処理手順

次の図に、ユーザーが要求を発行してから印刷されるまでの、印刷要求の流れを示します。

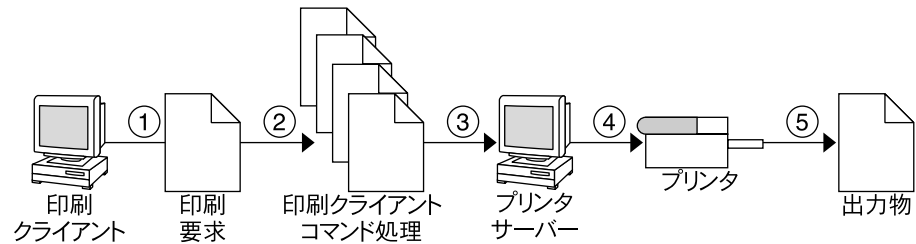


図 2-1 印刷クライアントの処理手順の概要

1. ユーザーは印刷クライアントから印刷要求を出します。
2. 印刷コマンドは印刷構成資源の階層をチェックして、印刷要求をどこに送信するか決定します。
3. 印刷コマンドは、印刷要求を適切なプリンタサーバーに直接送信します。プリンタサーバーは、BSD 印刷プロトコルを受け付ける任意のサーバーであり、SVR4 (LP) プリンタサーバーや BSD lpd ベースのプリンタサーバーなどがあります。
4. プリンタサーバーは印刷要求を適切なプリンタに送信します。
5. 印刷要求が印刷されます。

印刷クライアントの使用

この節では「印刷クライアント」、つまり印刷要求をプリンタサーバーに送信できるシステムと、印刷クライアントが印刷要求を発行するための印刷コマンドを中心に説明します。

図 2-2 に、印刷手順の中で、ユーザーが印刷クライアントから印刷要求を発行する処理を強調して示します。

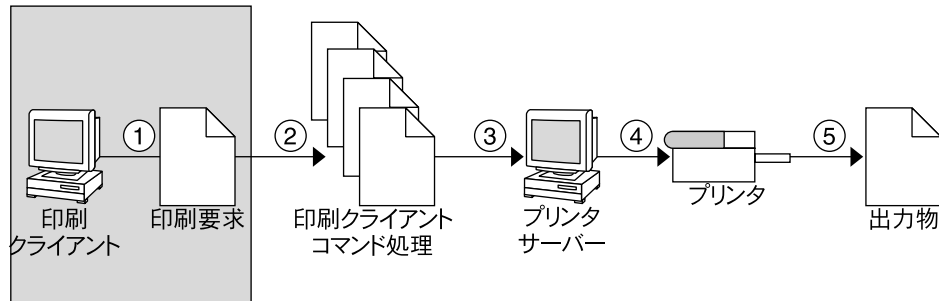


図 2-2 ユーザーが印刷クライアントから印刷要求を発行する

印刷クライアントとは

システムに Solaris 印刷ソフトウェアをインストールして、リモートプリンタにアクセスできるようにすると、そのシステムは印刷クライアントになります。

Solaris 印刷ソフトウェアは、次の資源からプリンタやプリンタ構成情報を見つけます。

- コマンド行インタフェースの `lp -d` コマンド (単独名形式または POSIX 形式)
- ユーザーの `LPDEST` 変数または `PRINTER` 変数
- `/etc/nsswitch.conf` ファイルに `printers` データベースとして設定されたソースの `_default` 変数
- ユーザーの `$HOME/.printers` ファイル
- ローカル `/etc/printers.conf` ファイル
- LDAP または NIS ネームサービスデータベース

印刷クライアントは、その要求をプリンタサーバーの待ち行列に送信します。つまり、クライアントは、ローカルの待ち行列を持ちません。クライアントが印刷要求を一時スプール領域に書き込むのは、プリンタサーバーが利用できない場合か、エラーが発生した場合だけです。サーバーまでの経路がこのような簡素化されているために、印刷クライアントは少ない資源で処理を実行でき、印刷障害の発生する可能性が減り、パフォーマンスが向上します。

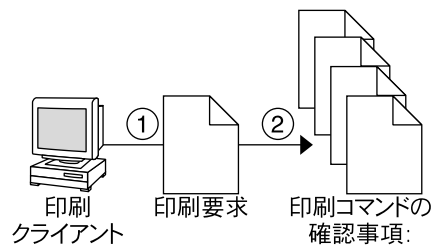
プリンタ構成資源

この節では、印刷ソフトウェアが、プリンタ名とプリンタ構成情報を見つけるのに使用する資源について説明します。

印刷ソフトウェアは、ネットワーク上のすべてのプリンタのプリンタ構成情報を格納するネットワーク (共有) 資源である、ネームサービスを使用できます。ネームサービス (LDAP、NIS、NIS+、または NIS+ (xfn)) は、プリンタ構成情報の管理を簡単にします。プリンタをネームサービスに追加すると、ネットワーク上のすべての印刷クライアントは、そのプリンタにアクセスできます。

印刷ソフトウェアがプリンタを見つける手順

次の図に示すように、印刷ソフトウェアは、プリンタとプリンタ構成情報を見つけるためにより多くのオプションを使用します。



- | | |
|--|---|
| A. 単独形式、POSIX形式、またはコンテキストベース形式のプリンタ名またはプリンタクラス | E. ローカルの <code>/etc/printers.conf</code> ファイル |
| B. デフォルトプリンタを指定するユーザーの <code>PRINTER</code> 環境変数または <code>LPDEST</code> 環境変数 | F. ドメインの 'ou=printers' コンテナにあるプリンタに関する LDAP ディレクトリ情報ツリー |
| C. <code>/etc/nsswitch.conf</code> にあるプリンタデータベース内の <code>_default</code> 変数 | G. NIS <code>printers.conf.byname</code> マップ |
| D. ユーザーの <code>\$HOME/.printers</code> ファイル | H. NIS+ <code>printers.org_dir</code> テーブル |
| | I. NIS+ (xfn) FNS 印刷コンテキスト |

図 2-3 印刷クライアントソフトウェアがプリンタを見つける手順

- ユーザーは `lp` コマンドまたは `lpr` コマンドを使用して、印刷クライアントから印刷要求を出します。ユーザーは、次の 3 つの形式のいずれかを使用して、宛先のプリンタ名またはプリンタクラスを指定できます。
 - 単独名形式。次の例に示すように、`lp` コマンドとオプションの後にプリンタ名またはプリンタクラスが続きます。

```
% lp -d neptune filename
```
 - POSIX 形式。次の例に示すように、印刷コマンドとオプションの後に `server:printer` が続きます。

```
% lpr -P galaxy:neptune filename
```
 - コンテキストベース形式。次の例に示します。

```
% lpr -d thisdept/service/printer/printer-name filename
```
- 印刷コマンドは、次の手順でプリンタとプリンタ構成情報を見つけます。

- ユーザーが宛先のプリンタ名またはプリンタクラスを3つの有効な形式のいずれかで指定しているかどうかをチェックします。
- ユーザーがプリンタ名またはプリンタクラスを有効な形式で指定していない場合、ユーザーの PRINTER 環境変数または LPDEST 環境変数にデフォルトプリンタ名が指定されていないか調べます。
- どちらの環境変数にもフォルトプリンタが指定されていない場合は、`/etc/nsswitch.conf`ファイルに`printers` データベースとして設定されたソースを調べます。ネームサービス資源は、次のうちの1つである可能性があります。
 - ドメインの `ou=printers` コンテナ内の LDAP ディレクトリ情報ツリー
 - NIS `printers.conf.byname` マップ
 - NIS+ `printers.conf_dir` マップ
 - NIS+ `xfn` (FNS) 印刷コンテキスト

プリンタサーバーの使用

プリンタサーバーはローカルプリンタが接続されているシステムであり、プリンタがネットワーク上の他のシステムを利用できるようにします。図 2-4 に、印刷手順の中で、プリンタサーバーが印刷要求をプリンタに送信する処理を強調して示します。

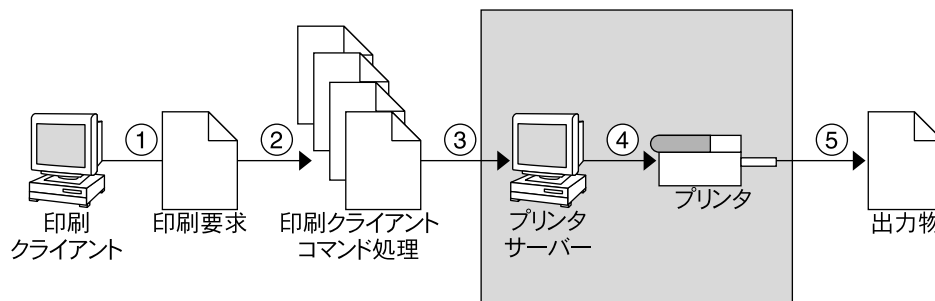


図 2-4 プリンタサーバーが印刷要求をプリンタに送信する

1. 印刷クライアントが印刷要求を発行する。
2. 印刷要求が印刷クライアント上で処理される。
3. 印刷要求がプリンタサーバーに送信される。
4. 印刷要求がプリンタに送信される。
5. プリンタから印刷出力される。

BSD 印刷プロトコル

印刷コマンドは、BSD 印刷プロトコルを使用します。このプロトコルの利点の1つは、さまざまなプリンタサーバーと通信できることです。

- `lpd` ベースのプリンタサーバー

- SunOS 5.9 プリンタサーバーおよび互換性のある SVR4 (LP) プリンタサーバー
- BSD 印刷プロトコルをサポートする、その他のプリンタサーバーやプリンタ

BSD 印刷プロトコルは、さまざまなメーカーの異なるシステム間の互換性を提供するため、業界で広く使用されている標準規格です。Sun は、将来の相互運用性を提供するために、BSD 印刷プロトコルをサポートします。

次に進む手順

Solaris プリンタマネージャを使用して新しいプリンタを設定する詳しい手順を知りたい場合は、第 4 章に進みます。計画を立てるための情報が必要な場合は、第 3 章を参照してください。

第 3 章

ネットワーク上のプリンタの計画方法 (概要)

この章では、最も効率よくネットワーク間でプリンタを分散する方法とプリンタの構成を計画する方法について説明します。

この章で説明する主な内容は次のとおりです。

- 42 ページの「ネットワーク上でのプリンタの分散」
- 42 ページの「プリンタサーバーと印刷クライアントを割り当てる」
- 43 ページの「プリンタサーバーの要件と推奨」

プリンタ関連作業の情報の参照箇所

プリンタ関連作業	参照箇所
Solaris プリンタマネージャと lp コマンドによるプリンタのセットアップ	第 4 章
セットアップ後の lp コマンドによるプリンタの管理	第 5 章
文字セット、フィルタ、フォーム、フォントの管理	第 6 章
LP 印刷サービスのカスタマイズ	第 7 章

ネットワーク上でのプリンタの分散

ネットワーク上のプリンタを設定する目的は、ユーザーが1つまたは複数のプリンタにアクセスできるようにすることです。管理者として、各プリンタを1台のシステム専用にするのが効率がよいか、多数のシステムが利用できるようにするのがよいかを判断しなければなりません。ネットワーク環境では、プリンタを複数のプリンタサーバーに分散させます。複数のプリンタサーバーを設定する利点は、あるプリンタサーバーに問題が発生しても、別のプリンタサーバーに印刷要求を振り替えられることです。

集中化した印刷構成を採用した場合も、使い勝手をよくしたり、応答時間を短縮したりするために、プリンタをユーザーのシステムに接続できます。ユーザーのシステムに接続されたプリンタも、ネットワーク上の他のシステムから利用できます。

次の図は、集中化印刷構成を採用した場合もプリンタをユーザーのシステムに接続できる例を示しています。

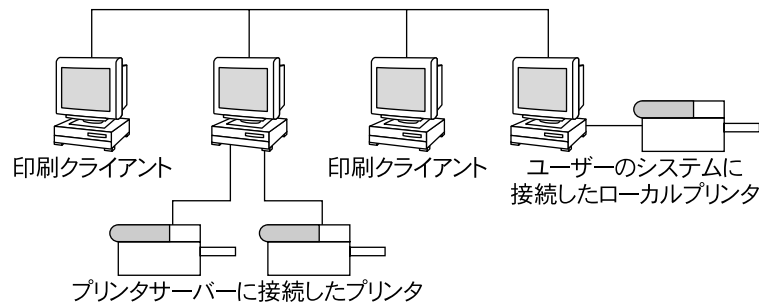


図 3-1 ネットワーク上でプリンタを分散する方法

プリンタサーバーと印刷クライアントを割り当てる

どのシステムにローカルプリンタを接続し、どのシステムでは他のシステム上のプリンタを使用するかを決定する必要があります。ローカルプリンタを接続し、そのプリンタをネットワーク経由で他のシステムでも利用できるようにしているシステムを「プリンタサーバー」と呼びます。プリンタサーバーに印刷要求を送るシステムを「印刷クライアント」と呼びます。

LP 印刷サービスは、Solaris 環境で印刷サービスを管理するソフトウェアです。プリンタをシステムに物理的に接続したら、LP 印刷サービスにプリンタサーバーの特性を定義しなければなりません。プリンタサーバーを設定し終わったら、他のシステムを印刷クライアントとして設定できます。

プリンタサーバーと印刷クライアントは、Solaris リリースのさまざまなバージョンと UNIX オペレーティングシステムのさまざまなバージョンで実行することができます。Solaris 9 リリースとその互換バージョンを実行する印刷クライアントは、SunOS 4.1、BSD UNIX、Linux などの lpd をベースにした印刷サービスを実行するプリンタサーバーに印刷することができます。さらに、lpd をベースにした印刷サービスを実行する印刷クライアントは、Solaris 9 リリースとその互換バージョンを実行するプリンタサーバーに印刷することができます。

以下の図は、Solaris リリースと lpd をベースにした印刷サービスを実行するシステムを備えたネットワークの印刷構成を示しています。Solaris の印刷機能が異機種システム混在環境でどのように動作するのかについては、175 ページの「リモート印刷の処理スケジュール」を参照してください。

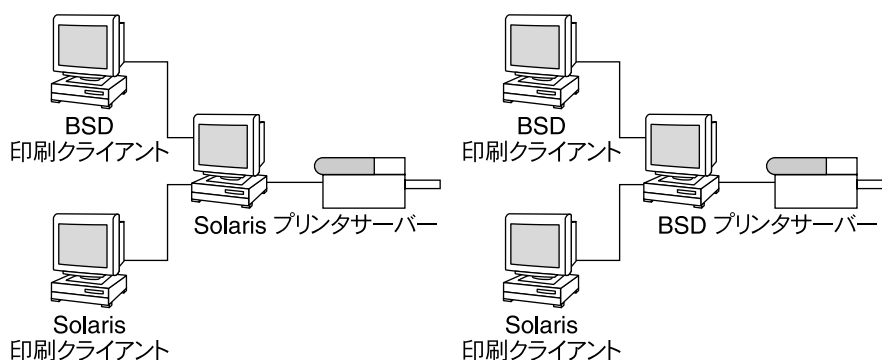


図 3-2 Solaris と lpd をベースにしたシステムの印刷構成

プリンタサーバーの要件と推奨

プリンタは、スタンドアロンシステムまたはネットワーク上の任意のシステムに接続できます。ネットワークに接続されていて、プリンタを備えているシステムは、印刷負荷を管理できる十分な資源があれば、どれでもプリンタサーバーとして使用できます。

スプーリング空間

「スプーリング空間」とは、印刷待ち行列内で要求を格納し、処理するためのディスク空間です。スプーリング空間は、どのシステムをプリンタサーバーとして指定するかを決定する場合の唯一重要な要素です。ユーザーがファイルの印刷要求を出すと、それらのファイルは印刷が完了するまで `/var/spool/lp` ディレクトリに格納されます。`/var` ディレクトリのサイズは、ディスクのサイズとディスクのパーティション分割方法によって異なります。スプーリング空間は、プリンタサーバーの `/var` ディレクトリに割り当てられるか、ファイルサーバーからマウントされてネットワーク上でアクセスされます。

注 - `/var` が別のスライスとして作成されていない場合、`/var` ディレクトリはルート (`/`) ファイルシステムの空間を使用します。これは、不十分なプリンタサーバーの空間になる可能性があります。

ディスク空間

システムをプリンタサーバーの候補として評価するときは、利用できるディスク空間を考慮してください。大きなスプールディレクトリでは、600M バイトのディスク空間を消費することもあります。プリンタサーバーとして指定できるシステム上で、ディスク空間のサイズと分割設定を調べてください。

また、印刷のニーズと印刷クライアントシステムの利用パターンを慎重に調べてください。小さなグループのユーザーが短い電子メールメッセージ、つまり複雑なフォームを必要としない単純な ASCII ファイルだけを印刷する場合は、20 ~ 25M バイトのディスク空間を `/var` ディレクトリに割り当てたプリンタサーバーで十分でしょう。ただし、多数の印刷クライアントユーザーが長い文書、ビットマップ、またはラスタイメージファイルを印刷する場合は、スプーリング空間が頻繁に一杯になります。ユーザーがジョブを印刷待ち行列に入れることができないと、作業の流れは中断してしまいます。より多くのスプーリング空間を要求すると、システム管理者はスプーリング用のディスク空間を追加するか、異なるシステムをグループのプリンタサーバーとして指定しなくてはならない場合があります。

プリンタサーバーの `/var` ディレクトリが小さすぎる場合で、より大きなファイルシステムが利用可能なときは、`/var` ディレクトリ上に大規模ファイルシステムをマウントしてスプーリング空間を追加することができます。ファイルシステムのマウントと `vfstab` ファイルの編集については、『Solaris のシステム管理 (基本編)』の「ファイルシステムのマウントとマウント解除 (手順)」を参照してください。

メモリー

Solaris 環境を実行するには、少なくとも 64M バイトのメモリーが必要です。プリンタサーバー用にメモリーを追加する必要はありません。ただし、メモリーが多ければ、印刷要求のフィルタリング処理が高速になります。

スワップ空間

プリンタサーバー上のスワップ空間は、LP 印刷サービスを処理するのに十分な量を割り当ててください。スワップ空間を増やす方法については、『Solaris のシステム管理 (基本編)』の「追加スワップ空間の構成 (手順)」を参照してください。

ハードディスク

最適のパフォーマンスを発揮させるには、プリンタサーバーにハードディスクとローカル /var ディレクトリが必要です。プリンタサーバー用のスプーリング空間をローカルのハードディスク上にマウントしてください。プリンタサーバーに専用のハードディスクとローカルの /var ディレクトリがあれば、印刷処理ははるかに高速になります。また、印刷要求の処理に必要な時間をより正確に予測できます。

プリンタ設定の計画

この節では、Solaris 環境で印刷するための計画の立て方の概要を説明します。

- プリンタ名、プリンタの説明、プリンタポートなどのプリンタの定義の設定
- プリンタタイプとファイル内容の形式の選択
- 障害通知とデフォルトプリンタの宛先の設定
- バナーページを設定するかどうか、あるいはプリンタへのユーザーのアクセスを制限するかどうかの決定
- プリンタクラスと障害回復の設定

プリンタの定義の設定

ネットワーク上でのプリンタの定義は、より効率的な印刷環境をユーザーに提供するための継続的な作業です。たとえば、サイトにあるすべてのプリンタのパラメータを設定すれば、ユーザーはプリンタがどこにあるのかを見つけやすくなります。あるいは、プリンタのクラスを定義することにより、印刷要求を迅速に処理できます。

lpadmin コマンドを使用すると、すべての印刷定義を設定できます。一方、Solaris プリンタマネージャを使用すると、プリンタのインストールまたは変更時に印刷定義の一部だけを設定できます。表 3-1 は、印刷定義と、その定義を Solaris プリンタマネージャで割り当てることができるかどうかを示しています。

表 3-1 Solaris プリンタマネージャで設定される印刷定義

印刷定義	Solaris プリンタマネージャで設定できるか
プリンタ名	設定できる
プリンタの説明	設定できる
プリンタポート	設定できる
プリンタタイプ	設定できる
ファイル内容形式	設定できる。ただし lpadmin コマンドより機能は少ない
障害通知	設定できる。ただし lpadmin コマンドより機能は少ない
デフォルトプリンタ	設定できる
バナーページの印刷	設定できる。ただし lpadmin コマンドより機能は少ない
プリンタへのユーザーアクセスの制限	設定できる。ただし lpadmin コマンドより機能は少ない
プリンタクラス	設定できない
障害回復	設定できない

プリンタ名

システムにプリンタを追加するときは、その「プリンタ名」を指定します。プリンタ名は、次の規則に従ってください。

- 管理ドメイン内のすべてのプリンタ間で一意であること
- 最大 14 文字までの英数字 (ハイフンと下線も含む) であること
- 覚えやすく、プリンタのタイプ、場所、プリンタサーバー名などを識別できること

サイトに合った命名規則を設定してください。たとえば、ネットワーク上で異なるタイプのプリンタを使用する場合は、プリンタ名の一部にプリンタタイプを含めると、ユーザーは適切なプリンタを選択しやすくなります。たとえば、PostScript プリンタは文字 `ps` で識別できます。ただし、サイトのプリンタがすべて PostScript プリンタである場合は、`ps` というイニシャルをプリンタ名の一部として含める必要はありません。

プリンタの説明

`lpadmin -D` コマンドまたは Solaris プリンタマネージャを使用すると、プリンタに説明を割り当てることができます。プリンタに割り当てる説明には、ユーザーがプリンタを識別できるような情報を含める必要があります。プリンタが設置されている部屋番号、プリンタのタイプやメーカー、印刷に問題がある場合に連絡する担当者名などを含めることができます。

次のコマンドでプリンタの説明を調べることができます。

```
$ lpstat -D -p printer-name
```

プリンタポート

プリンタのインストール時、またはその設定を後から変更するときに、`lpadmin -p printer-name -v device-name` コマンドまたは Solaris プリンタマネージャを使用して、プリンタの接続先となるデバイス、つまり「プリンタポート」を指定できます。

ほとんどのシステムが、2つのシリアルポートと1つのパラレルポートまたはUSBポートを持っています。システムにポートを追加しないかぎり、3台以上のシリアルプリンタおよび2台以上のパラレルプリンタまたは3台以上のUSBプリンタを直接接続することはできません。

Solaris プリンタマネージャを使用すると、以下のプリンタポートタイプを選択することができます。これらのオプションには、`lpadmin` コマンドと同じ柔軟性があります。

プリンタポートタイプ	対応するデバイス名のオプション
シリアル	<code>/dev/term/a</code>
シリアル	<code>/dev/term/b</code>
パラレル	<code>/dev/printers/0 -> /dev/ecpp0</code>
USB	<code>/dev/printers/[1-9]</code>
プリンタサーバーが認識するポート名を指定	Other

LP 印刷サービスは、標準プリンタインタフェースプログラムからの設定を使用してプリンタポートを初期設定します。プリンタインタフェースプログラムの詳細は、114 ページの「印刷フィルタの管理」を参照してください。デフォルト設定で機能しないパラレルプリンタやシリアルプリンタがある場合は、133 ページの「プリンタポート特性の調整」のポート設定をカスタマイズする方法を参照してください。

IA のみ - IA システムで複数のポートを使用している場合、デフォルトでは最初のポートだけが有効です。2 番目以降のポートはデフォルトでは無効です。複数のポートを使用するためには、追加の `asy` (シリアル) ポートや `lp` (パラレル) ポートごとに、デバイスドライバのポート構成ファイルを手作業で編集しなければなりません。IA ポート構成ファイルのパスは、次のとおりです。

```
/platform/i86pc/kernel/drv/asy.conf
```

```
/platform/i86pc/kernel/drv/lp.conf
```

プリンタタイプ

プリンタタイプとは、プリンタの種類を表す一般名です。プリンタタイプは、プリンタのさまざまな制御シーケンスが入っているterminfo データベースエントリを識別します。通常、プリンタタイプはメーカーのモデル名からとります。たとえば、DECwriter™のプリンタタイプ名は decwriter です。ただし、共通プリンタタイプ PS はこの規則に従いません。PS は Apple LaserWriter® I や Apple LaserWriterII プリンタなど、多くのPostScript™ プリンタモデルのプリンタタイプとして使用されます。

lpadmin -T コマンドまたは Solaris プリンタマネージャを使用すると、プリンタタイプを指定できます。

Solaris プリンタマネージャを使用すると、メニューからプリンタタイプを選択するか、「その他 (Other)」を選択して terminfo データベース内でプリンタタイプを指定できます。この方法には、lpadmin コマンドと同じ機能があります。

terminfo データベース内のプリンタ名

各プリンタタイプに関する情報は、terminfo データベース (/usr/share/lib/terminfo) に格納されています。この情報には、プリンタの機能と初期制御データが含まれます。インストールするプリンタは、terminfo データベース内のエントリに対応していなければなりません。

```
$ pwd
/usr/share/lib/terminfo
$ ls
1 3 5 7 9 B H P a c e g i k m o q s u w y
2 4 6 8 A G M S b d f h j l n p r t v x z
$
```

各サブディレクトリには、端末またはプリンタに関してコンパイル済みのデータベースエントリが入っています。各エントリは、プリンタまたは端末のタイプの頭文字別に編成されています。たとえば、Epson® プリンタがある場合は、/usr/share/lib/terminfo/e ディレクトリ内を探すと、Epson プリンタの特定のモデルが見つかります。

```
$ cd /usr/share/lib/terminfo/e
$ ls
emots          ep2500+high  ep48          ergo4000      exidy2500
env230         ep2500+low  epson2500     esprit
envision230   ep40        epson2500-80  ethernet
ep2500+basic  ep4000     epson2500-hi  ex3000
ep2500+color  ep4080     epson2500-hi80 exidy
$
```

上記のように、Epson プリンタのエントリがあります。

NEC® プリンタがある場合は、/usr/share/lib/terminfo/n ディレクトリ内を探すと、使用中の NEC プリンタモデルが見つかります。


```

$ cd /usr/share/lib/terminfo/n
$ ls
ncr7900          ncr7900iv      netronics      network        nuc
ncr7900-na      ncr7901       netty          netx           nuclterm
ncr7900i        nec            netty-Tabs    newhp
ncr7900i-na     net           netty-vi      newhpkeyboard
$

```

上記のように、このディレクトリには、NECのエントリが含まれています。

プリンタタイプの選択

ローカル PostScript プリンタの場合は、プリンタタイプとして PostScript (PS) または Reverse PostScript (PSR) を使用します。使用するプリンタが PostScript をサポートしていれば、プリンタタイプが terminfo データベースに含まれていても、PS または PSR を選択してください。

PostScript プリンタでページの印刷面を上にして印刷すると、文書は逆方向に印刷されます。1 ページ目はスタックの 1 番下になり、最終ページは 1 番上になります。プリンタのタイプを PSR として指定すると、LP 印刷サービスはプリンタに送る前にページの順序を逆転させます。つまり、最終ページが最初に印刷され、各ページは正順にスタックされます。ただし、LP 印刷サービスがページ順を確実に変更できるのは、『PostScript リファレンスマニュアル (PostScript Language Reference Manual)』(Adobe Systems Incorporated 制作、Addison-Wesley 社、1990 年刊) の付録 C の Adobe® Document Structuring Conventions に準拠する PostScript ファイルの場合だけです。

プリンタで複数の種類のプリンタをエミュレートできる場合は、lpadmin -T コマンドを使用して複数のタイプを割り当てることができます。複数のプリンタタイプを指定すると、LP 印刷サービスは各印刷要求に適したタイプを使用します。

該当する terminfo ディレクトリ内でプリンタタイプが見つからないことがあります。プリンタのタイプは、そのプリンタのメーカー名に対応しているとは限りません。たとえば、PostScript プリンタのタイプによっては、メーカーや製品名に固有のエントリの代わりに、PS または PSR エントリ (/usr/share/lib/terminfo/P ディレクトリに入っています) を使用できます。

例外的なタイプのプリンタを使用する場合は、さまざまなエントリを試してみなければ、プリンタのモデルに使用できる特定の terminfo エントリを判断できないことがあります。できれば、プリンタに使用できるエントリを terminfo データベース内で見つけてください。既存のエントリを使用するほうが、新たにエントリを作成するよりはるかに簡単です。独自のエントリを作成しなければならない場合は、135 ページの「サポートされていないプリンタの terminfo エントリを追加する」を参照してください。役立つヒントが掲載されています。

ファイル内容形式の選択

印刷フィルタはファイルの内容を、目的のプリンタが受け付けることができる形式に変換します。「ファイル内容形式」は、フィルタを通さずに直接印刷できるファイル内容の形式を LP 印刷サービスに通知します。フィルタなしに印刷するには、必要なフォントをプリンタ上でも利用できなければなりません。フィルタは、その他のファイル形式用に設定して使用します。

lpadmin -I コマンドまたは Solaris プリンタマネージャを使用すると、プリンタのファイル内容形式を指定できます。Solaris プリンタマネージャを使用すると、メニューからファイル内容形式を選択できます。一部のファイル内容形式はメニューにありません。lpadmin コマンドを使用して、Solaris プリンタマネージャメニューにないファイル内容形式を指定してください。

ほとんどのプリンタは、以下のファイルタイプを直接印刷することができます。

- プリンタタイプと同じタイプ。たとえば、PostScript プリンタ用であれば ps
- simple タイプ。たとえば、ASCII テキストファイル

ユーザーがファイルの印刷要求を出すときは、lp -T *content-type* コマンドを使用してそのファイルの内容形式を指定することができます。要求を出すときにファイルの内容形式を指定しないと、LP サーバーは要求の最初のファイルを見て内容形式を判定します。ファイルが ^D%! または %! で始まっている場合、要求には PostScript™ データが含まれると見なされます。それ以外の場合、ファイルは simple (ASCII) テキストと見なされます。LP 印刷サービスはファイル内容形式を使用して、ファイル内容をプリンタで処理できる形式に変換するためのフィルタを決めます。

Solaris プリンタマネージャではファイル内容形式のリストが表示されるので、ローカルプリンタをインストールまたは変更するときに、そこから形式を選択できます。選択結果は LP 印刷サービスが使用する名前に変換されます。次の表は、Solaris プリンタマネージャで選択できるファイル内容形式を示しています。

表 3-2 Solaris プリンタマネージャによるファイル内容形式の選択

ファイル内容形式	LP 印刷サービス名	説明
PostScript	postscript	PostScript ファイルはフィルタを通す必要がない
ASCII	simple	ASCII ファイルはフィルタを通す必要がない
PostScript と ASCII	simple, postscript	PostScript ファイルも ASCII ファイルもフィルタを通す必要がない
なし	" "	プリンタのタイプに一致するもの以外は、すべてのファイルがフィルタを通す必要がある

表 3-2 Solaris プリンタマネージャによるファイル内容形式の選択 (続き)

ファイル内容形式	LP 印刷サービス名	説明
任意	any	フィルタは使用されない。プリンタがファイル内容形式を直接処理できなければ、そのファイルは印刷されない

プリンタの機能に最も適合するファイル内容形式を選択してください。PostScript は、Solaris プリンタマネージャのデフォルトの選択で、通常はほとんどこのまま使用できます (PostScript ファイルには、フィルタ処理が不要なことを示します)。

通常使用するプリンタ

この節では、Solaris ソフトウェアで最も一般的に使用されるプリンタのプリンタタイプとファイル内容形式について説明します。掲載されていませんが、ここで説明するプリンタの多くは、simple 内容形式のファイルも直接印刷できます。

PostScript プリンタがある場合は、プリンタタイプ PS または PSR と内容形式 postscript を使用してください。PSR はページの順序を逆転させ、各ページを逆順で印刷してバナーページを最後に印刷します。

表 3-3 は、PostScript 以外の他のプリンタと、各プリンタの構成に使用するプリンタタイプを示しています。これらのプリンタでは、ファイル内容形式は simple です。

注 - Sun では表 3-3 のプリンタをサポートしていませんが、フィルタ処理を行うか、プリンタがファイル内容形式を直接印刷できれば、サポートしていないプリンタを使用できます。以下の製品に不明な点がある場合は、製造元に問い合わせてください。

表 3-3 Sun がフィルタを提供していない PostScript 以外のプリンタ

プリンタ	プリンタタイプ
Daisy	daisy
Datagraphix	datagraphix
DEC LA100	la100
DEC LN03	ln03
DECwriter	decwriter
Diablo	diablo diablo-m8
Epson 2500 系列	epson2500

表 3-3 Sun がフィルタを提供していない PostScript 以外のプリンタ (続き)

プリンタ	プリンタタイプ
	epson2500-80
	epson2500-hi
	epson2500-hi80
Hewlett-Packard HPCL printer	hplaser
IBM Proprinter	ibmproprinter

terminfo データベースにないプリンタを設定したい場合は、138 ページの「サポートされていないプリンタの terminfo エントリを追加する方法」を参照してください。

第 4 章

プリンタの設定手順 (手順)

この章では、プリンタを設定し、Solaris プリンタマネージャでネットワーク上のシステムにアクセスできるようにする方法を説明します。

Solaris プリンタマネージャを使用したプリンタの設定手順については、53 ページの「印刷の設定 (作業マップ)」を参照してください。

プリンタの概要については、第 2 章を参照してください。

印刷の設定 (作業マップ)

作業	説明	参照先
1. 新しく接続したプリンタの追加	Solaris プリンタマネージャを使用する - プリンタをシステムに物理的に接続したあとで、プリンタを印刷できるようにする	59 ページの「新しく接続するプリンタを Solaris プリンタマネージャを使用して追加する方法」
2. プリンタへのアクセスの追加	Solaris プリンタマネージャを使用する - 印刷クライアントでプリンタへのアクセスを追加	61 ページの「Solaris プリンタマネージャにプリンタアクセスを追加する方法」
3. (省略可能) .printers ファイルの設定	\$HOME/.printers ファイルを使用して、ユーザーが独自のカスタムプリンタの別名を設定できるようにする	63 ページの「.printers ファイルを設定する方法」

作業	説明	参照先
4. 新しいネットワークプリンタの追加	<p>プリンタのベンダーが提供するツールを使用する – プリンタをネットワークに物理的に接続した後、ベンダーが提供するソフトウェアを使用してネットワークプリンタを構成する</p> <p>LPコマンドを使用する – プリンタをネットワークに物理的に接続した後、ネットワークプリンタを構成する</p>	<p>68 ページの「プリンタベンダー提供のツールを使用してネットワークプリンタを追加する方法」</p> <p>69 ページの「LP コマンドを使用してネットワークプリンタを追加する方法」</p>
5. (省略可能) バナーページの抑制	バナーページが印刷されないように抑制できる	85 ページの「バナーページを抑制する方法」
6. (省略可能) 障害警告の設定	Solaris プリンタマネージャよりも具体的な障害警告を設定できる	88 ページの「プリンタの障害警告を設定する方法」
7. (省略可能) 障害回復の設定	プリンタの障害後の回復方法を設定できる	90 ページの「プリンタの障害回復を設定する方法」
8. (省略可能) プリンタへのアクセス制限	Solaris プリンタマネージャは許可リストを設定できるが、プリンタにアクセスするユーザーを制限したい場合は、拒否リストを設定する	92 ページの「プリンタへのユーザーアクセスを制限する方法」

Solaris プリンタマネージャによる印刷の設定

Solaris プリンタマネージャは Java ベースのグラフィカルユーザーインターフェースで、ローカルおよびリモートのプリンタ構成を管理できます。このツールは、ネームサービス環境 (LDAP、NIS、NIS+、フェデレーテッド・ネーミング・サービス (xfn) を使用した NIS+、およびファイル) 内で使用できます。このツールを使用するには、スーパーユーザーとしてログインする必要があります。

Solaris プリンタマネージャを使用すると、プリンタサーバー (「新しいローカルプリンタを設定 (New Attached Printer)」) と印刷クライアント (「プリンタへのアクセスを追加 (Add Access to Printer)」) を設定できます。ローカルプリンタまたは接続プリンタとは、プリンタサーバーに物理的にケーブル接続されたプリンタのことです。ネットワークプリンタとは、ネットワークに物理的に接続されたプリンタのことです。プリンタへのアクセスを追加する (つまり、リモートアクセスを追加する) とは、印刷クライアント (サーバー以外のすべてのマシン) がプリンタにアクセスできるようにする手順のことです。

Solaris プリンタマネージャによるプリンタの設定

次の表はプリンタ属性を示しています。Solaris プリンタマネージャを使用してプリンタを設定するときに、必要となる情報を判断するのに役立ててください。

プリンタ属性	説明	例	デフォルト設定	必須か省略可能か
プリンタ名	プリンタの名前	laser1	なし	接続されたプリンタやネットワークプリンタをインストールし、プリンタへのアクセスを追加するのに必須
プリンタサーバー	プリンタサーバーの名前	venus	ローカルシステム	接続されたプリンタやネットワークプリンタをインストールし、プリンタへのアクセスを追加するのに必須
説明	ユーザー定義の文字列	laser printer near breakroom	なし	省略可能
プリンタポート	デバイスプリンタが接続されているポート	/dev/term/a	/dev/term/a	接続されたプリンタをインストールするのに必須
プリンタタイプ	プリンタのタイプ	unknown	PostScript	接続されたプリンタやネットワークプリンタをインストールするのに必須
ファイル内容	印刷する内容	any	PostScript	接続されたプリンタやネットワークプリンタをインストールするのに必須
宛先	ネットワークプリンタの宛先名	例については、66 ページの「宛先 (またはネットワークプリンタアクセス) 名の選択」を参照	なし	ネットワークプリンタをインストールするのに必須
プロトコル	プリンタとの通信に使用するプロトコル	TCP	BSD	ネットワークプリンタをインストールするのに必須
失敗の通知	ユーザーにエラーを知らせる方法	Mail to superuser	Write to superuser	省略可能
デフォルトプリンタ	デフォルトプリンタを識別する	なし	なし	省略可能
バナーを常に印刷	バナーが印刷される	なし	バナーが印刷される	省略可能

プリンタ属性	説明	例	デフォルト設定	必須か省略可能か
ユーザーアクセスリスト	印刷できるユーザーのリスト	rimmer, lister	すべてのユーザーが印刷できる	省略可能

Solaris プリンタマネージャと特殊文字

Solaris プリンタマネージャは、入力画面のさまざまなテキストフィールドへのユーザー入力をチェックします。チェックの方法は2種類あります。一般的に不当な入力と特定のフィールドで不当な入力がないかどうかをチェックします。

Solaris プリンタマネージャは、ヘルプ画面以外の入力で、次の文字を受け付けません。

- "\$^&*(){}' | ; : ? < > "などのシェルのメタ文字。コロンを受け付けるネットワークプリンタ画面の出力先フィールドは除く
- 複数バイト文字
- #、スペース、タブ。タブを受け付ける記述フィールドは除く

Solaris プリンタマネージャの起動

Solaris プリンタマネージャを使用してプリンタを設定するには、CDE ワークスペースメニューから「プリンタ管理 (Printer Administration)」を選択して Solaris プリンタマネージャを起動するか、コマンド行から Solaris プリンタマネージャを起動します。詳細については、次の節を参照してください。

▼ Solaris プリンタマネージャを起動する方法

1. 次の前提条件を満たしていることを確認します。Solaris プリンタマネージャを使用するには、次の条件を満たす必要があります。
 - ビットマップディスプレイモニターがある。Solaris プリンタマネージャは、Sun ワークステーションの標準ディスプレイモニターなど、ビットマップ画面のコンソールを使用するシステムでだけ使用できます。
 - CDE 環境のような X Window System を実行しているか、xhost 環境が動作するシステムでリモート表示機能を使用している。
 - 接続されたプリンタやネットワークプリンタをインストールする場合は、プリンタサーバーにスーパーユーザーとしてログインしている。プリンタへのアクセスを追加する場合は、印刷クライアントにスーパーユーザーとしてログインしている。
 - LDAP、NIS、NIS+、または NIS+ (xfn) データベースを管理するのに必要なアクセス特権を持っている。

- ネームサービスが LDAP の場合は、次の事項が必要になります。
 - プリンタの管理者の識別名 (DN) とディレクトリのパスワード
 - 管理ドメインにサービスを提供する LDAP サーバーの IP アドレスまたは名前。通常は、自動的に表示されて選択されます。
LDAP を使用したプリンタ情報の管理については、28 ページの「LDAP によるプリンタ情報の管理」を参照してください。
- ネームサービスが NIS の場合は、NIS マスターの root パスワードが必要です。
- ネームサービスが NIS+ の場合は、次の手順を実行しなければならない場合があります。
 - a. NIS+ マスターにスーパーユーザーとしてログインします。
 - b. 次に示すようにプリンタテーブルを所有するグループを確認します。

```
# niscat -o printers.org_dir.domain_name.com
.
.
.
Group          : "admin.domain_name.com"
```

- c. 必要なら、`printers.org_dir.<domain>` ファイルの更新を許可された NIS+ admin グループに、Solaris プリンタマネージャを実行するシステムを追加します。


```
# nisgrpadm -a admin.domain_name.com host_name
```
 - d. Solaris プリンタマネージャを実行するシステムにスーパーユーザーとしてログインします。NIS+ 構成によっては、`/usr/bin/keylogin` コマンドも実行しなければならない場合があります。詳細については、`keylogin(1)` を参照してください。
- ネームサービスが NIS+ (xfn) の場合は、次の手順を実行しなければならない場合があります。
 - a. NIS+ マスターにスーパーユーザーとしてログインします。
 - b. フェデレーテッド・ネーミングテーブルを所有するグループを確認します。

```
# niscat -o fns.ctx_dir.domain_name.com
.
.
.
Group          : "admin.domain_name.com"
```

- c. 必要なら、`fns.ctx_dir.<domain>` ファイルの更新を許可された NIS+ admin グループに、Solaris プリンタマネージャを実行するシステムを追加します。


```
# nisgrpadm -a admin.domain_name.com host_name
```
- d. Solaris プリンタマネージャを実行するシステムにスーパーユーザーとしてログインします。NIS+ 構成によっては、`/usr/bin/keylogin` コマンドも実行しなければならない場合があります。詳細については、`keylogin(1)` を参照してください。

- SUNWppm パッケージをインストールします。

```
# pkginfo | grep SUNWppm
system      SUNWppm      Solaris Print Manager
```

2. 次のいずれかの方法で、Solaris プリントマネージャを起動します。

- CDE のワークスペースメニューの「ツール (Tools)」オプションから「プリンタ管理 (Printer Administrator)」を選択します。
- CDE フロントパネルから「アプリケーション (Applications)」メニューを選択し、アプリケーション (Application Manager) の「システム管理 (System_Admin)」ウィンドウで「プリンタ管理 (Printer Administrator)」アイコンをクリックします。
- コマンド行から、次のように入力します。

```
# /usr/sadm/admin/bin/printmgr &
```

Solaris プリントマネージャのメインウィンドウ上に「ネームサービスを選択 (Select Naming Service)」ウィンドウが重なって表示されます。

リモートシステムから Solaris プリントマネージャを使用したい場合は、次のように行います。

ローカルシステムで `xhost` コマンドを使用し、リモートシステム表示用のアクセスを与えるには、次のように入力します。

```
# xhost +remote-system
```

次に、リモートシステムにログインして、`DISPLAY` 環境変数を設定し、Solaris プリントマネージャを起動します。

```
# DISPLAY=local-system:display_number
# export DISPLAY
# /usr/sadm/admin/bin/printmgr &
```

注 - CDE メニューやコマンド行から Solaris プリントマネージャを起動できない場合は、次の確認をしてください。

- a. ローカルシステムやリモートシステムの Xserver プロセスに接続する権限がスーパーユーザー (root) にない可能性があります。その場合は、次のように入力します。

```
$ xhost +hostname
$ su
(Enter root's password)
# /usr/sadm/admin/bin/printmgr &
```

Solaris プリントマネージャを再起動する前に、ローカルシステムまたはリモートシステムの名前で `hostname` を置き換えます。

- b. ローカルシステムまたはリモートシステムに SUNWppm パッケージがインストールされていることを確認します。

```
$ pkginfo | grep SUNWppm
```

3. ネットワークで使用されているネームサービスを「ネームサービスを選択 (Select Naming Service)」ウィンドウから選択します。選択肢には **NIS+ (xfn)**、**NIS+**、**NIS**、**files** があります。
4. ドメイン名が正しいことを確認します。
ネームサービスが正常に読み込まれると、Solaris プリンタマネージャのメインウィンドウが表示されます。

プリンタサーバーの設定

接続されたプリンタとネットワークプリンタ、またはそのどちらかをシステムに追加すると、そのプリンタにローカルシステムからアクセスできるようになります。プリンタをインストールするシステムが「プリンタサーバー」になります。

次の各項では、Solaris プリンタマネージャを使用して、直接接続されているプリンタやネットワークプリンタをプリンタサーバーに追加する方法を説明します。Solaris プリンタマネージャを使用する手順に続く例は、LP コマンドを使用してプリンタを追加する方法です。

▼ 新しく接続するプリンタを Solaris プリンタマネージャを使用して追加する方法

1. プリンタサーバーであるシステムを選択します。
2. プリンタをプリンタサーバーに接続し、プリンタの電源を入れます。
ハードウェアのスイッチとケーブル接続の要件については、プリンタのインストールマニュアルを参照してください。
3. プリンタを接続したプリンタサーバー上で **Solaris** プリンタマネージャを起動します。
手順については、56 ページの「Solaris プリンタマネージャを起動する方法」を参照してください。
4. 「プリンタ (Printer)」メニューから「新しいローカルプリンタを設定 (New Attached Printer)」を選択します。
「新しいローカルプリンタを設定 (New Attached Printer)」ウィンドウが表示されます。
5. ウィンドウに情報を入力します。
フィールドに入力するために情報が必要な場合は、「ヘルプ (Help)」ボタンをクリックします。

6. 「了解 (OK)」をクリックします。
7. プリンタがインストールされていることを確認します。確認するには、**Solaris** プリンタマネージャのメインウィンドウに新しいプリンタエントリがあるか調べます。
8. プリンタが要求どおり印刷できることを確認します。

```
$ lp -d printer-name filename
```

9. **Solaris** プリンタマネージャを終了します。
「プリンタマネージャ (Print Manager)」メニューから「終了 (Exit)」を選択します。

例 — LP コマンドを使用して新規に接続したプリンタを追加する

この例では、ローカルの PostScript プリンタをプリンタサーバーで印刷できるようにする方法を示しています。この例のコマンドは、プリンタが接続されているプリンタサーバーで実行しなければなりません。この例では次の情報を使用しています。実際に指定する情報はこれとは異なります。

- プリンタ名: luna
- ポートデバイス: /dev/term/b
- プリンタタイプ: PS
- ファイル内容形式: postscript

```
# chown lp /dev/term/b
# chmod 600 /dev/term/b 1
# lpadmin -p luna -v /dev/term/b 2
# lpadmin -p luna -T PS 3
# lpadmin -p luna -I postscript 4
# cd /etc/lp/fd
# for filter in *.fd;do
  > name=`basename $filter .fd`
  > lpfilter -f $name -F $filter
  > done 5
# accept luna
  destination "luna" now accepting requests
# enable luna 6
printer "luna" now enabled
# lpadmin -p luna -D "Room 1954 ps" 7
# lpstat -p luna 8
printer luna is idle. enabled since Jul 12 11:17 2001. available.
```

1. lp に所有権とポートデバイスへの単独アクセスを設定する。
2. プリンタ名とプリンタが使用するポートデバイスを定義する。
3. プリンタのプリンタタイプを設定する。
4. プリンタが直接印刷できるファイル内容形式を指定する。
5. 印刷フィルタをプリンタサーバーに追加する。
6. プリンタが印刷要求を受け入れて、印刷できるようにする。
7. プリンタの説明を追加する。

8. プリンタが用意できていることを確認する。

次に進む手順

次の表を見て、次に進む手順を決めてください。

作業	参照箇所
印刷クライアントに新しくインストールしたプリンタへのアクセスを追加する (プリンタ情報をネームサービスデータベースに追加していない場合)	61 ページの「Solaris プリンタマネージャにプリンタアクセスを追加する方法」
.printers ファイルを設定する	63 ページの「.printers ファイルを設定する方法」

印刷クライアントの設定

印刷クライアントは、プリンタ用のサーバーではないが、プリンタにアクセスできるシステムのことです。印刷クライアントは、プリンタサーバーのサービスを使用して、印刷ジョブのプール、スケジュール、およびフィルタリングを実行します。1つのシステムがあるプリンタのプリンタサーバーになり、同時に他のプリンタの印刷クライアントになることも可能です。

プリンタへのアクセスは、ドメイン全体で有効にすることも、マシン単位で有効にすることもできます。これは、プリンタ情報をネームサービスデータベースに追加するかどうかによって異なります。

次の項では、新しい Solaris プリンタマネージャを使用して、印刷クライアントでプリンタへのアクセスを追加する方法を説明します。手順に続く例は、LP コマンドを使用してプリンタアクセスを追加する方法です。

▼ Solaris プリンタマネージャにプリンタアクセスを追加する方法

1. リモートプリンタへのアクセスを追加したいシステム上で **Solaris** プリンタマネージャを起動します。
手順については、56 ページの「Solaris プリンタマネージャを起動する方法」を参照してください。
2. 「プリンタ (**Printer**)」メニューから「プリンタへのアクセスを追加 (**Add Access to Printer**)」の順に選択します。

「プリンタへのアクセスを追加 (Add Access to Printer)」ウィンドウが表示されま
す。

3. ウィンドウに情報を入力します。
フィールドに入力するための情報が必要な場合は、「ヘルプ (Help)」ボタンをク
リックします。
4. 「了解 (OK)」をクリックします。
5. プリンタへのアクセスが追加されていることを確認します。確認するには、**Solaris** プ
リントマネージャのメインウィンドウに新しいプリンタエントリがあるか調べます。
6. プリンタが要求どおり印刷できることを確認します。

```
$ lp -d printer-name filename
```

7. **Solaris** プリントマネージャを終了します。
「プリンタマネージャ (Print Manager)」メニューから「終了 (Exit)」を選択します。

例 — LP コマンドでプリンタアクセスを追加する

リモートプリンタで印刷する場合、リモートプリンタにアクセスを追加しなければな
りません。次の例は、プリンタサーバー saturn のプリンタ luna にアクセスを構成
する方法を示しています。システム saturn は、プリンタ luna の印刷クライアント
になります。

```
# lpadmin -p luna -s saturn 1  
# lpadmin -p luna -D "Room 1954 ps" 2  
# lpadmin -d luna 3  
# lpstat -p luna 4
```

1. プリンタとプリンタサーバーを指定する。
2. プリンタの説明を追加する。
3. プリンタをシステムのデフォルトのプリンタ宛先として設定する。
4. プリンタが用意できていることを確認する。

.printers ファイルの設定

プリンタ情報をカスタマイズする必要がなければ、ユーザーのホームディレクトリに
.printers ファイルを設定する必要はありません。しかし、.printers ファイル
は、ユーザーが独自のプリンタ別名を設定できる場所です。別名 `_default` を使用す
れば、デフォルトのプリンタを設定できます。また、特殊別名 `all` を設定すれば、
印刷要求を取り消したりプリンタの情報をチェックしたりするときの対象となるプリ
ンタのリストを定義できます。

LP 印刷サービスが `.printers` ファイルを使用するかどうかは、ネームサービススイッチ (`/etc/nsswitch.conf`) によって制御されます。デフォルト構成では、印刷サービスは、ユーザーのホームディレクトリでプリンタ構成情報を探してから他のネームサービスを調べます。つまり、ユーザーのプリンタ構成ファイルを設定することによって、ネームサービスの共有情報ではなく、好みのプリンタ情報を使用できます。

`.printers` ファイルの詳細については、`printers(4)` のマニュアルページを参照してください。ネームサービススイッチの詳細については、`nsswitch.conf(4)` のマニュアルページを参照してください。

▼ `.printers` ファイルを設定する方法

1. スーパーユーザーとしてシステムにログインします。
2. 任意のエディタで、`.printers` ファイルをユーザーのホームディレクトリに作成します。
3. (省略可能) 次の例で示すように入力して、別名 `_default` を設定し、指定のプリンタをデフォルトにします。

```
_default printer_name
```

4. (省略可能) 別名 `_all` を設定して、印刷要求を取り消したりプリンタの状態をチェックしたりするときの対象となるプリンタを定義します。次の例に示すようなエントリを使用します。

```
_all printer1,printer2,printer3
```

5. そのファイルを `.printers` として保存します。

ネットワークプリンタの追加

「ネットワークプリンタ」とは、ネットワークに直接接続されているハードウェアデバイスです。これは、ネットワークプリンタがプリンタサーバーにケーブルで実際に接続されていなくても、プリンタサーバーからアクセスできることを意味します。ネットワークプリンタは専用のシステム名と IP アドレスを持っています。ネットワークプリンタがプリンタサーバーに接続されていない場合でも、専用のプリンタサーバーを設定しておく必要があります。プリンタサーバーは、ネットワークプリンタの待ち行列化機能と印刷管理機能を提供します。

ネットワークプリンタは、ベンダー提供の印刷プログラムを必要とする特別なプロトコルを1つ以上使用することがあります。ベンダーから提供される印刷プログラムの設定手順は、それぞれ異なることがあります。プリンタにベンダー提供サポートが付いていない場合、ほとんどのデバイスについて Solaris のネットワークプリンタサポートを使用できます。可能な場合は、プリンタのベンダーが提供するソフトウェアを使用します。

ベンダーは、SVR4 プリンタインタフェーススクリプトを提供して標準プリンタインタフェーススクリプトを置き換えている場合があります。その場合、SVR4 インタフェーススクリプトはベンダー提供の印刷プログラムを呼び出して、ジョブをプリンタに送ります。このスクリプトが提供されない場合は、標準インタフェーススクリプトを変更してベンダー提供の印刷プログラムを呼び出す必要があります。この作業は、標準インタフェーススクリプトのプリンタごとのコピーをベンダー提供の印刷プログラムを呼び出すように編集することで実行できます。

ネットワークプリンタ構成で使用する用語を説明します。

- プリンタサーバー – プリンタのジョブをスプールおよびスケジュールするマシン。このマシンにプリンタが構成されます。
- プリンタホストデバイス – プリンタホストデバイスは、ベンダーが提供するソフトウェアとハードウェアです。ネットワークに対応していないプリンタにネットワークプリンタの機能をサポートします。プリンタホストデバイスとそれに接続された1つまたは複数のプリンタの組み合わせは「ネットワークプリンタ」になります。
- プリンタノード – 物理的なプリンタまたはプリンタホストデバイスのいずれかのことです。ネットワークサポートが物理的なプリンタにあるときは、物理的なプリンタです。ネットワークインタフェースを提供するために外部ボックスを使用しているときは、プリンタホストデバイスです。プリンタノード名は、IPアドレスが与えられているマシン名です。この名前はシステム管理者が選択するもので、デフォルトやベンダーの要件はありません。ノードと同様に、プリンタノード名も一意でなければなりません。
- プリンタ名 – 何らかのプリンタのコマンドを使用する時に、コマンド行に入力する名前。この名前は、システム管理者がプリンタ構成時に選択します。1つの物理的なプリンタは、複数のプリンタ名または待ち行列名を持つことができます。それぞれ、そのプリンタへのアクセスを提供します。
- 宛先またはネットワークプリンタのアクセス名 – プリンタのサブシステムがプリンタへのアクセス用に使用するプリンタノードポートの内部名。プリンタノード名か、プリンタベンダーポート指定付きのプリンタノード名です。プリンタベンダーポート指定は、プリンタベンダーのマニュアルで明示的に定義されています。これはプリンタに固有です。プリンタがプリンタホストデバイスでありプリンタでもある場合、ポート指定は、プリンタホストデバイスのマニュアルに説明されています。書式は次のいずれかです。

printer_node_name

または

printer_node_name:port_designation

- プロトコル–プリンタとのケーブル経由の通信に使用するプロトコル。プリンタのマニュアルには、選択するプロトコルについての説明があります。ネットワークプリンタサポートは、BSD プリンタプロトコルと raw TCP の両方を提供します。実装によって、両方を使用するように設定できます。
- タイムアウト (再試行間隔)–タイムアウトはプリンタへの接続試行の間の待機秒数をあらわすシード (seed) 数です。このシード数は、接続の試行間に待機する最小の秒数であり、接続が失敗するごとに増えます。プリンタへの接続が繰り返して失敗すると、ユーザーの介入を要求するメッセージがユーザーに戻されます。接続が成功するか、ジョブの所有者がジョブを取り消すまで、再接続の試行は続けられません。

ネットワークプリンタ用のプリンタベンダー提供のソフトウェア

一般に、ネットワークプリンタには、プリンタベンダーから提供されているソフトウェアが必要です。プリンタにベンダー提供のソフトウェアがある場合、可能ならば、プリンタベンダーのソフトウェアを使用します。プリンタベンダー提供のソフトウェアは、そのプリンタの特性をサポートするように設計されていて、プリンタの能力を最大限に活用します。プリンタのマニュアルをよく読んで、プリンタを LP 印刷システムにインストールおよび構成してください。

Sun のネットワークプリンタのサポート

ネットワークプリンタベンダーがソフトウェアサポートを提供していない場合、Sun が提供するソフトウェアを利用できます。このソフトウェアは、ネットワークプリンタの汎用サポートを提供するもののため、必ずしもプリンタで利用できるすべての機能を使用できません。

ネットワークプリンタを追加するための一般的な説明については、第 4 章を参照してください。次は、Sun 提供のソフトウェアを使用したプリンタの管理を説明します。

ネットワークプリンタサポートの呼び出し

ネットワークプリンタ用のソフトウェアサポートは、インタフェーススクリプト経由で呼び出されます。ネットワークインタフェーススクリプト `netstandard` でネットワークプリンタを構成すると、ネットワークプリンタサポートモジュールが呼び出されます。次に、ネットワークサポートでプリンタを構成するコマンドを示します。

```
lpadmin -p printer_name -m netstandard
```

プロトコルの選択

印刷サブシステムは、BSD 印刷プロトコルと raw TCP を使用してプリンタと通信します。プリンタのマニュアルには、使用するプロトコルについての情報が提供されています。一般に、プリンタに使用するのは TCP プロトコルです。

プロトコルを選択するコマンドは次のいずれかです。

```
lpadmin -p printer_name -o protocol=bsd
```

または

```
lpadmin -p printer_name -o protocol=tcp
```

選択したプロトコルが BSD 印刷プロトコルの場合、さらにコントロールファイルをプリンタに送信する順番を選択できます。一部のプリンタは、コントロールファイルの後にデータファイルという順番を仮定しますが、その逆を仮定するプリンタもあります。詳しくは、プリンタベンダーのマニュアルを参照してください。デフォルトでは、コントロールファイルを先に送信します。

順番を選択するコマンドは次のいずれかです。

```
lpadmin -p printer_name -o bsdctrl=first
```

または

```
lpadmin -p printer_name -o bsdctrl=last
```

プリンタノード名の選択

システム管理者はプリンタノード名を選択します。ネットワーク上のノードと同様に、この名前は一意でなければなりません。プリンタノード名は、プリンタの IP アドレスと関連付けられます。

宛先 (またはネットワークプリンタアクセス) 名の選択

印刷サブシステムはプリンタのアクセス情報を必要とします。これは、プリンタへのネットワーク接続を行うときにサブシステムが使用する名前です。この名前は、システム管理者が lpadmin コマンドで印刷サブシステムに提供します。これは、プリンタ構成データベースの一部になります。プリンタアクセス名はプリンタノード名であり、ポート名で修飾される場合もあります。ポート指定はプリンタベンダー間で異なります。ポート指定については、プリンタのマニュアルを参照してください。次に、プリンタアクセス名の書式を示します。

```
printer_node-name [: port_designation]
```

例 1 — ポート指定 (番号) 付き宛先名 (またはネットワークプリンタアクセス名)

TCP の共通ポート指定は 9100 です。プリンタノード名が `pn1` で、プリンタベンダーがそのポートを 9100 と定義していた場合、プリンタアクセス名は `pn1:9100` になります。この場合にプリンタを構成するには、次のコマンドを使用します。

```
lpadmin -p printer_name -o dest=pn1:9100
```

例 2 — ポート指定 (名前) 付き宛先名 (またはネットワークプリンタアクセス名)

BSD プロトコルを使用するとき、ポート指定は番号でなく、プリンタベンダーが定義した名前です (例: `xxx_parallel_1`)。プリンタノード名が `cardboard` の場合、プリンタアクセス名は `cardboard:xxx_parallel_1` になります。この場合にプリンタを構成するには、次のコマンドを使用します。

```
lpadmin -p printer_name -o dest=cardboard:xxx_parallel_1
```

例 3 — ポート指定なしの宛先名 (またはネットワークプリンタアクセス名)

ポート指定がなく、プリンタノード名が `newspaper` の場合、プリンタアクセス名はプリンタノード名 `newspaper` になります。この場合にプリンタを構成するには、次のコマンドを使用します。

```
lpadmin -p printer_name -o dest=newspaper
```

タイムアウト値の設定

タイムアウトオプションは、プリンタに接続しようとする試行間で待機する時間 (秒数) を個別に選択するためのものです。ウォームアップ時間が長いプリンタの場合は、タイムアウト値を大きくします。デフォルトは 10 秒です。

タイムアウト値は、印刷プロセスが成功するか失敗するかには影響を与えません。これは、ソフトウェアが初期タイムアウトカウントとして使用するシード値です。失敗が続くと、このカウントは増えます。プリンタへの接続の試行が連続して失敗すると、メッセージがスプーラに送信されます。これによって、ユーザーの介入が必要であることをユーザーに警告します。プリンタの電源が入っていなかったり、用紙がなくなっていたりするときにも、このメッセージが生成される可能性があります。たとえば、プリンタがウォームアップしているときに、このようなメッセージが頻繁に生成されるようであれば、タイムアウト値を増やすことで間違ったメッセージを減らすことができます。

システム管理者は最適なタイムアウト値を探してください。次に、タイムアウト値を設定するコマンドを示します。

```
lpadmin -p printer_name -o timeout=n
```

ネットワークプリンタアクセスの管理

各ネットワークプリンタは、そのプリンタへのアクセスを提供するサーバーを1つだけ持っています。これによって、サーバーはそのプリンタへのアクセスを管理して、ジョブの一貫性を保つことができます。

ネットワークプリンタのデフォルトデバイスは /dev/null です。プリンタに待ち行列が1つしかない場合はこれで十分です。複数の待ち行列が必要であれば、そのデバイスをファイルに設定します。これによって、印刷システムはプリンタへのアクセスを待ち行列間で制限できます。次のコマンドは、デバイスファイルを作成して、ネットワークプリンタデバイスとして構成しています。

```
touch /path/filename
chmod 600 /path/filename
lpadmin -p printer_name -v /path/filename
```

次の例では、devtreedown というデバイスファイルを作成しています。

```
# touch /var/tmp/devtreedown
# chmod 600 /var/tmp/devtreedown
# lpadmin -p treedown -v /var/tmp/devtreedown
```

▼ プリンタベンダー提供のツールを使用してネットワークプリンタを追加する方法

1. プリンタをネットワークに接続して、プリンタの電源を入れます。
ハードウェアのスイッチとケーブル接続の要件については、プリンタのインストールマニュアルを参照してください。IP アドレスを取得して、プリンタノード名を選択します。これは、ネットワークにノードを追加することと同じです。
2. プリンタのマニュアルに従って、ネットワークプリンタを追加してください。
プリンタのマニュアルを使用して、ネットワークプリンタを構成してください。手順は、ベンダーやプリンタに固有です。
3. 新しいプリンタへのアクセスをクライアントに追加します。
これでプリンタは追加されました。プリンタへのアクセスをクライアントに作成します。詳細は、61 ページの「印刷クライアントの設定」を参照してください。
4. オプションの作業を完了します。
ネットワークプリンタを設定するときは、オプションの作業がいくつかあります。残りの作業については、53 ページの「印刷の設定 (作業マップ)」を参照してください。

▼ LP コマンドを使用してネットワークプリンタを追加する方法

ここでは、ネットワークプリンタサポートソフトウェアを使用して、ネットワークプリンタを設定するのに必要な手順を説明しています。このソフトウェアを使用するのは、プリンタにベンダー提供のソフトウェアが付いていない場合だけです。

1. プリンタをネットワークに接続して、プリンタの電源を入れます。
ハードウェアのスイッチとケーブル接続の要件については、プリンタのインストールマニュアルを参照してください。IP アドレスを取得して、プリンタノード名を選択します。これは、ネットワークにノードを追加することと同じです。
2. ネットワークプリンタを構成するのに必要な情報を収集します。
 - プリンタ名
 - プリンタサーバー
 - ネットワークプリンタアクセス名
 - プロトコル
 - タイムアウト

詳細は、63 ページの「ネットワークプリンタの追加」で説明されている用語を参照してください。

3. プリンタ名、デバイス、プリンタタイプ、内容形式を定義します。
 - a. プリンタ名とプリンタが使用するポートデバイスを定義します。

```
# lpadmin -p printer-name -v /dev/null
使用するデバイスは /dev/null です。
```

- b. プリンタが使用するインタフェースを指定します。

```
# lpadmin -p printer-name -m /netstandard
ネットワークプリンタサポートソフトウェアで提供されるインタフェーススクリプトは /usr/lib/lp/model/netstandard です。
```

- c. プリンタ宛先、プロトコル、およびタイムアウト値を設定します。

```
# lpadmin -p printer-name -o dest=access-name:port -o protocol=protocol -o timeout=value
```

`-p printer-name`

ネットワークプリンタ名を指定する

`-o dest= access-name:port`

ネットワークプリンタアクセス名と、プリンタのマニュアルに定義されていれば指定されたプリンタベンダーポートに、プリンタ宛先を設定する。

<code>-o protocol=protocol</code>	プリンタとケーブル経由で通信するために使用するプロトコルを設定する。BSD と raw TCP の両方をサポートしている
<code>-o timeout=value</code>	プリンタへの接続の試行間で待機する秒数を表す再試行タイムアウト値を設定する。

d. プリンタのファイル内容形式とプリンタタイプを指定します。

```
# lpadmin -p printer-name -I content-type -T printer-type
```

詳細については、lpadmin(1M) のマニュアルページを参照してください。

4. フィルタをプリンタサーバーに追加します。

```
# cd /etc/lp/fd
# for filter in *.fd;do
> name=`basename $filter .fd`
> lpfilter -f $name -F $filter
> done
```

5. プリンタがプリンタ要求を受け入れて、その要求を印刷できるようにします。

```
# accept printer-name
# enable printer-name
```

6. プリンタが正しく構成されていることを確認します。

```
# lpstat -p printer-name
```

7. 新しいプリンタへのアクセスをクライアントに追加します。

これでプリンタは追加されました。プリンタへのアクセスをクライアントに作成します。詳細は、61 ページの「印刷クライアントの設定」を参照してください。

8. オプションの作業を完了します。

プリンタを設定するときは、オプションの作業がいくつかあります。残りの作業については、53 ページの「印刷の設定 (作業マップ)」を参照してください。

例 — LP コマンドを使用してネットワークプリンタを追加する

この例のコマンドは、プリンタサーバーで実行する必要があります。この例では次の情報を使用しています。実際の状況に合わせて指定する情報を変更してください。

- プリンタ名: luna1
- サーバー: saturn
- ネットワークプリンタアクセス名: nimquat:9100
- プロトコル: tcp
- タイムアウト:5

- インタフェース: /usr/lib/lp/model/netstandard
- プリンタタイプ: PS
- 内容形式: postscript
- デバイス: /dev/null

```
# lpadmin -p luna1 -v /dev/null 1
# lpadmin -p luna1 -m netstandard 2
# lpadmin -p luna1 -o dest=nimquat:9100 -o protocol=tcp
-o timeout=5 3
# lpadmin -p luna1 -I postscript -T PS 4
# cd /etc/lp/fd
# for filter in *.fd;do
  > name=`basename $filter .fd`
  > lpfilter -f $name -F $filter
  > done 5
# accept luna1
  destination "luna1" now accepting requests
# enable luna1 6
printer "luna1" now enabled
# lpadmin -p luna1 -D "Room 1954 ps" 7
# lpstat -p luna1 8
printer luna1 is idle. enabled since Jul 12 11:17 2001. available.
```

1. プリンタ名を定義する。デバイスを /dev/null に設定する。
2. ネットワークプリンタ用のインタフェーススクリプトを定義する。
3. 宛先、プロトコル、およびタイムアウトを設定する。
4. プリンタが直接印刷できるファイル内容形式とプリンタタイプを指定する。
5. 印刷フィルタをプリンタサーバーに追加する。
6. プリンタが印刷要求を受け入れて、印刷できるようにする。
7. プリンタの説明を追加する。
8. プリンタが用意できていることを確認する。

プリンタ構成情報を変換する

この節では、サイトにある SunOS 5.5.1 またはその互換バージョンを稼動しているシステムのプリンタ構成情報を変換して、その情報を印刷クライアントにコピーすることにより、既存のプリンタにアクセスできるようにする方法を説明します。

注 - 既存のプリンタの台数が少ない場合は、Solaris プリンタマネージャか Admintool を使用してプリンタへのアクセスを追加する方が、プリンタ構成情報を変換して印刷クライアントに配布するよりも簡単です。プリンタへのアクセスの追加については、53 ページの「印刷の設定 (作業マップ)」を参照してください。

プリンタ構成情報を変換する (作業マップ)

次の作業マップに、プリンタ構成情報を変換する作業の概要を示します。

作業	説明	参照先
既存のプリンタ構成情報を変換する	<p>SunOS 5.5.1 リリースを実行しているシステムのプリンタ構成情報を変換する - SunOS 5.5.1 リリースを使用しているサイトでは、 /etc/lp/printers ディレクトリのプリンタ構成情報を /etc/printers.conf 構成ファイルに変換する。通常、この作業は1回行うだけでよい</p> <p>lpd ベースのプリンタサーバーを実行するシステムのプリンタ構成情報を変換する - lpd ベースのソフトウェアを使用しているサイトでは、 /etc/printcap ファイルのプリンタ構成情報を /etc/printers.conf 構成ファイルに変換する。通常、この作業は1回行うだけでよい</p>	<p>73 ページの「Solaris 2.5.1 システムのプリンタ情報を変換する方法」</p> <p>73 ページの「lpd ベースのプリンタサーバーを実行するシステムのプリンタ情報を変換する方法」</p>

既存のプリンタ構成情報の変換

既存のプリンタ構成情報は、Solaris 9 またはその互換バージョンをインストールあるいはアップグレードするときに自動的に変換されます。この節では、SunOS 5.5.1 リリースあるいは lpd ベースのプリンタサーバーを実行するシステムのプリンタ構成情報を /etc/printers.conf プリンタ構成ファイルに変換する方法を説明します。変換作業を自動化するために、2つの印刷管理コマンドのいずれかを使用します。

- `conv_lp` コマンドは、SunOS 5.9 システムの /etc/lp/printers ディレクトリの情報を、そのシステムの /etc/printers.conf ファイルのエントリに変換します。手順については、73 ページの「Solaris 2.5.1 システムのプリンタ情報を変換する方法」を参照してください。
- `conv_lpd` コマンドは、lpd ベースのプリンタサーバーを実行するシステムの /etc/printcap 構成ファイルの情報を、/etc/printers.conf ファイルのエントリに変換します。手順については、73 ページの「lpd ベースのプリンタサーバーを実行するシステムのプリンタ情報を変換する方法」を参照してください。

ネームサービスを使用していない場合、サイトにある既存のプリンタを含む、`/etc/printers.conf` のマスターファイルを作成します。次に、このマスターファイルをすべての印刷クライアントにコピーするか、(ネームサービスを使用している場合は)、NIS または NIS+ にロードします。新しい印刷クライアントは、最初から、サイトにある既存のプリンタにアクセスできるようにしておくことを推奨します。



注意 – NIS または NIS+ ネームサービスを使用してプリンタ情報を構成している場合、印刷クライアント上の `/etc/printers.conf` ファイルは使用しないでください。印刷クライアントは、最初に `/etc/printers.conf` ファイルを使用してプリンタを検出します。そのときに、`/etc/printers.conf` ファイル内の情報と、NIS マップまたは NIS+ マップ内のプリンタ情報に矛盾がある場合は、予期せぬ結果が生じることがあります。この問題を回避するために、印刷クライアントが NIS ネームサービスまたは NIS+ を使用してプリンタ情報を構成するときは、印刷クライアント上の `/etc/printers.conf` ファイルを削除してください。

▼ Solaris 2.5.1 システムのプリンタ情報を変換する方法

1. **Solaris 9** を実行しているシステムに、スーパーユーザーとしてログインします。
2. そのシステムの `/etc/lp/printers` ディレクトリ内のプリンタ構成情報を `/etc/printers.conf` ファイルに変換します。

```
# /usr/lib/print/conv_lp
```

▼ lpd ベースのプリンタサーバーを実行するシステムのプリンタ情報を変換する方法

1. lpd ベースのプリンタサーバーを実行するシステムの `/etc/printcap` ファイルを、**Solaris 9** リリースを実行するシステムにコピーします。
2. `/etc/printcap` ファイルのコピー先である **Solaris 9** を実行しているシステムに、スーパーユーザーとしてログインします。
3. `/etc/printcap` ファイル内のプリンタ構成情報を `/etc/printers.conf` ファイルに変換します。

```
# /usr/lib/print/conv_lpd
```


第 5 章

プリンタの管理 (手順)

この章では、LP コマンドでプリンタを管理する手順について説明します。

この章で説明する手順は次のとおりです。

- 76 ページの「プリンタとリモートプリンタへのアクセスを削除する方法」
- 79 ページの「プリンタの状態をチェックする方法」
- 80 ページの「印刷スケジューラを停止する方法」
- 81 ページの「印刷スケジューラを再起動する方法」
- 81 ページの「プリンタ記述を追加する方法」
- 83 ページの「システムのデフォルトプリンタを設定する方法」
- 84 ページの「バナーページを選択可能にする方法」
- 85 ページの「バナーページを抑制する方法」
- 87 ページの「プリンタのクラスを定義する方法」
- 88 ページの「プリンタの障害警告を設定する方法」
- 90 ページの「プリンタの障害回復を設定する方法」
- 92 ページの「プリンタへのユーザーアクセスを制限する方法」
- 94 ページの「印刷要求の状態をチェックする方法」
- 98 ページの「プリンタへの印刷要求を受け付けるまたは拒否する方法」
- 96 ページの「プリンタを使用可能または使用不可にする方法」
- 99 ページの「印刷要求を取り消す方法」
- 100 ページの「特定のユーザーからの印刷要求を取り消す方法」
- 101 ページの「印刷要求を別のプリンタに移動する方法」
- 103 ページの「印刷要求の優先順位を変更する方法」

印刷と LP 印刷サービスの概要については、第 2 章を参照してください。

プリンタと印刷スケジューラの管理

この節では、プリンタと印刷スケジューラを管理するために日常的に行う作業について説明します。

プリンタとプリンタアクセスの削除

プリンタの交換が必要な場合や、プリンタを別の場所に移動したい場合は、プリンタサーバーから物理的に削除する前に、LP 印刷サービスからプリンタ情報を削除しなければなりません。また、プリンタ上の現在の印刷要求がすべて印刷されるか、別のプリンタに移動して印刷されるかを確認する必要があります。

プリンタ情報をプリンタサーバーから削除するだけでなく、印刷クライアントまたはネットワークネームサービスからも削除する必要があります。プリンタサーバーからローカルプリンタを削除する場合は、印刷クライアントまたはネットワークネームサービスからリモートプリンタエントリを削除する必要があります。プリンタを別のプリンタサーバーに移動する場合は、印刷クライアントまたはネットワークネームサービスから古いリモートプリンタエントリを削除する必要があります。そうしてから、リモートプリンタの移動先へのアクセスを追加します。

ローカルとリモートのプリンタの削除方法については、76 ページの「プリンタとリモートプリンタへのアクセスを削除する方法」を参照してください。Solaris プリンタマネージャを使用して、ローカルプリンタまたはリモートプリンタを削除できます。ただし、Solaris プリンタマネージャでは、待ち行列に入っている印刷要求を別のプリンタに移動できません。

▼ プリンタとリモートプリンタへのアクセスを削除する方法

プリンタをサービスから削除するとき、プリンタアクセスを印刷クライアントから削除するには、次の手順を使用します。そうしてから、プリンタ情報をプリンタサーバーから削除します。

1. 削除したいプリンタへアクセスできる印刷クライアントに、スーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. 印刷クライアントからプリンタに関する情報を削除します。

```
print-client# lpadmin -x printer-name
```

`-x` 指定したプリンタを削除する
`printer-name` 削除したいプリンタ名

指定したプリンタに関する情報が、印刷クライアントの `/etc/lp/printers` ディレクトリから削除されます。

3. 印刷クライアントが同じプリンタサーバー上の別のプリンタを使用しない場合は、そのプリンタサーバーに関する情報を印刷クライアントから削除します。

```
print-client# lpsystem -r print-server
```

`-r` 指定したプリンタサーバーを削除する
`print-server` 削除したいプリンタサーバー名

プリンタサーバーが、印刷クライアントの `/etc/lp/Systems` ファイルから削除されます。

4. プリンタへアクセスできる各印刷クライアント上で、手順 2 から 手順 3 までを繰り返します。
5. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
6. プリンタ上で印刷要求を停止します。

```
print-server# reject printer-name
```

`reject printer-name` 指定したプリンタの印刷要求を拒否する

この手順を実行すると、プリンタの削除処理中は、そのプリンタの待ち行列に新しい要求が入らなくなります。詳細については、98 ページの「プリンタへの印刷要求を受け付けるまたは拒否する方法」を参照してください。

7. プリンタを停止します。

```
print-server# disable printer-name
```

この手順を実行すると、印刷要求の印刷は停止されます。印刷を停止する方法については、96 ページの「プリンタを使用可能または使用不可にする方法」を参照してください。

8. 待ち行列に残っている印刷要求がある場合は、別のプリンタに移動します。
印刷要求を別のプリンタに移動する方法については、101 ページの「印刷要求を別のプリンタに移動する方法」を参照してください。

9. プリンタをプリンタサーバーから削除します。

```
print-server# lpadmin -x printer-name
```

プリンタの構成情報が、プリンタサーバーの /etc/lp/printers ディレクトリから削除されます。

10. 削除したばかりのプリンタを使用していた印刷クライアントが、まだプリンタサーバー上で別のプリンタを使用中でなければ、その印刷クライアントに関する情報を削除します。

```
print-server# lpsystem -r print-client1 [,print-client2...]
```

-r 指定したプリンタサーバーを削除する

print-client プリンタサーバーから削除したい印刷クライアント名。このコマンドで複数の印刷クライアントを指定できる。印刷クライアント名を区切るには空白またはコンマを使用する。空白を使用する場合は、印刷クライアントのリストを引用符で囲む

指定した印刷クライアントが、プリンタサーバーの /etc/lp/Systems ファイルから削除されます。

11. プリンタ情報が削除されていることを確認します。

- a. 印刷クライアント上でプリンタ情報が削除されていることを確認します。

```
print-client$ lpstat -p printer-name -l
```

上記のコマンドの出力で、プリンタが存在しないことを示すエラーが表示されず。

- b. プリンタサーバー上でプリンタ情報が削除されていることを確認します。

```
print-server$ lpstat -p printer-name -l
```

上記のコマンドの出力で、プリンタが存在しないことを示すエラーが表示されず。

例 — プリンタとリモートプリンタへのアクセスを削除する

次の例は、印刷クライアント terra とプリンタサーバー jupiter からプリンタ luna を削除する方法を示しています。この例では、プリンタサーバーから印刷クライアント terra を削除する方法も示しています。

```
terra# lpadmin -x luna
Removed "luna".
terra# lpstat -p luna -l
jupiter# lpadmin -x luna
jupiter# lpsystem -r terra
Removed "terra".
```

```
jupiter# lpstat -p luna -l
```

プリンタの状態のチェック

多くの日常的なプリンタ管理作業には、LP 印刷サービスや特定のプリンタの状態に関する情報が必要です。たとえば、どのプリンタが使用できるかを判断し、そのプリンタの特性を検査しなければならない場合があります。lpstat コマンドを使用すると、LP 印刷サービスや特定のプリンタに関する状態情報を調べることができます。

▼ プリンタの状態をチェックする方法

1. ネットワーク上の任意のシステムにログインします。
2. プリンタの状態をチェックする
ここでは、最も一般的に使用するオプションだけを掲載してあります。その他のオプションについては、lpstat(1) を参照してください。

```
$ lpstat [-d] [-p printer-name [-D] [-l]] [-t]
```

-d	システムのデフォルトプリンタが表示される
-p <i>printer-name</i>	プリンタが使用可能かアイドル状態か、いつ使用可能または使用不可になったか、および印刷要求を受け付けているかどうかが表示される このコマンドで複数のプリンタ名を指定できる。プリンタ名を区切るには空白またはコンマを使用する。空白を使用する場合は、印刷クライアントのリストを引用符で囲む。 <i>printer-name</i> を指定しなければ、すべてのプリンタの状態が表示される
-D	指定した <i>printer-name</i> の記述が表示される
-l	指定した <i>printer-name</i> の特性が表示される
-t	すべてのプリンタの状態、使用可能かどうか、印刷要求を受け付けているかどうかなど、LP 印刷サービスに関する状態情報が表示される

例 — プリンタの状態をチェックする

次の例は、システムのデフォルトプリンタを表示する方法を示しています。

```
$ lpstat -d  
system default destination: luna
```

次の例は、プリンタ luna の状態を表示する方法を示しています。

```
$ lpstat -p luna
printer luna is idle. enabled since Jul 12 11:17 2001. available.
```

次の例は、プリンタ asteroid とプリンタ luna の記述を表示する方法を示しています。

```
$ lpstat -p "asteroid luna" -D
printer asteroid faulted. enabled since Jul 12 11:35 2001. available.
unable to print: paper misfeed jam
```

```
Description: Printer by break room
printer luna is idle. enabled since Jul 12 11:36 2001. available.
Description: Printer by server room.
```

次の例は、プリンタ luna の特性を表示する方法を示しています。

```
$ lpstat -p luna -l
printer luna is idle. enabled since Mon Jul 12 15:02:32 ...
  Form mounted:
  Content types: postscript
  Printer types: PS
  Description:
  Connection: direct
  Interface: /usr/lib/lp/model/standard
  After fault: continue
  Users allowed:
    (all)
  Forms allowed:
    (none)
  Banner not required
  Character sets:

  Default pitch:
  Default page size: 80 wide 66 long
  Default port settings:
```

印刷スケジューラの再起動

印刷スケジューラ lpsched は、プリンタサーバー上の印刷要求を処理します。ただし、印刷スケジューラがシステム上で動作を停止したために、印刷要求の受け付けや印刷が停止されることがあります。

次の節では、印刷スケジューラの再起動の方法について説明します。印刷スケジューラが動作を停止するときに印刷要求が印刷中だった場合は、印刷スケジューラを再起動すると、その印刷要求全体が印刷されます。

▼ 印刷スケジューラを停止する方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。

- 印刷スケジューラが動作しているかどうかをチェックします。

```
# lpstat -r
```

印刷スケジューラが動作していない場合は、「scheduler is not running」というメッセージが表示されます。

- 印刷スケジューラが動作している場合は停止します。

```
# /etc/init.d/lp stop
```

▼ 印刷スケジューラを再起動する方法

- プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。

- 印刷スケジューラが動作しているかどうかをチェックします。

```
# lpstat -r
```

印刷スケジューラが動作していない場合は、「scheduler is not running」というメッセージが表示されます。

- 印刷スケジューラが動作していない場合は起動します。

```
# /etc/init.d/lp start
```

その他のプリンタ定義の設定とリセット

この節では、プリンタ定義の設定またはリセットの手順を説明します。次のプリンタ定義の一部は、Solaris プリンタマネージャを使用して設定できます。次の手順では、迅速にプリンタ定義を設定またはリセットするための、lp コマンドの使用方法を示しています。

▼ プリンタ記述を追加する方法

- プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。

- プリンタを説明する記述を追加します。

```
# lpadmin -p printer-name -D "comment"
```

<code>-p printer-name</code>	記述を追加するプリンタ名
<code>-D "comment"</code>	設置場所や管理担当者など、プリンタの特性を指定する。シェルが解釈する文字 (*、?、\、!、^ など) は、一重引用符で囲む。

詳細については、`lpadmin(1M)` のマニュアルページを参照してください。
プリンタ記述はプリンタサーバーの `/etc/lp/printers/printer-name/comment` ファイルに追加されます。

3. Description 情報をチェックします。

```
$ lpstat -p printer-name -l
```

例 — プリンタ記述を追加する

次の例は、プリンタ `luna` のプリンタ記述を追加する方法を示しています。

```
# lpadmin -p luna -D "Nathans office"
```

デフォルトプリンタの指定

印刷コマンドを使用するときにプリンタ名を入力しなくてもすむように、ユーザーのデフォルトプリンタを指定できます。あるプリンタをデフォルトとして指定する前に、そのプリンタをシステム上の印刷サービスに認識させなければなりません。次のいずれかを設定すれば、ユーザーのデフォルトプリンタを設定できます。

- `LPDEST` 環境変数
- `PRINTER` 環境変数
- ユーザーの `.PRINTERS` ファイルの `_default` 変数
- システムのデフォルトプリンタ (`lpadmin -d` コマンドまたは Solaris プリンタマネージャを使用)

アプリケーションがプリンタを指定する場合は、システムのデフォルトプリンタを設定したかどうかに関係なく、その出力先が印刷サービスに使用されます。アプリケーションにプリンタの出力先がない場合や、印刷コマンドの使用時にプリンタ名が指定されていない場合は、印刷コマンドはデフォルトプリンタを特定の順序で検索します。表 5-1 は、システムのデフォルトプリンタの検索順序を示しています。

表 5-1 デフォルトプリンタの検索順序

検索順序	/usr/bin/lp コマンドを使用	lpd ベースの互換コマンド (lpr、lpq および lprm) を使用
1	LPDEST 変数	PRINTER 変数
2	PRINTER 変数	LPDEST 変数
3	システムのデフォルトプリンタ	システムのデフォルトプリンタ

▼ システムのデフォルトプリンタを設定する方法

1. デフォルトプリンタを設定したいシステムに、スーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. システムのデフォルトプリンタを設定します。

```
# lpadmin -d [printer-name]
```

`-d printer-name` システムのデフォルトプリンタとして割り当てるプリンタ名。
`printer-name` を指定しなければ、システムはデフォルトプリンタなしで設定される

デフォルトプリンタ名がシステムの `/etc/lp/default` ファイルに入力されます。

3. システムのデフォルトプリンタをチェックします。

```
$ lpstat -d
```

例 — システムのデフォルトプリンタを設定する

次の例は、プリンタ luna をシステムのデフォルトプリンタとして設定する方法を示しています。これは、LPDEST または PRINTER 環境変数が設定されていない場合、luna がシステムのデフォルトプリンタとして使用されることを意味します。

```
# lpadmin -d luna
# lpstat -d
system default destination: luna
```

バナーページの印刷

バナーページには、印刷要求を出したユーザー、印刷要求 ID、要求の印刷時期が出力されます。また、バナーページには、ユーザーがプリントアウトを識別しやすいように変更可能なタイトルを付けることもできます。

バナーページは、印刷ジョブの所有者を簡単に識別できるようにします。これは、多数のユーザーが同じプリンタにジョブを依頼するときに特に便利です。ただし、バナーページを印刷すると用紙の消費量が増えますが、1台のプリンタを使用するユーザーが少ない場合は必要ないことがあります。また場合によっては、バナーページを印刷しない方がよいこともあります。たとえば、プリンタに支払い小切手などの特殊な用紙やフォームが装着されている場合は、バナーページを印刷すると問題が起きることがあります。

デフォルトでは、印刷サービスはバナーページを強制的に印刷します。ただしユーザーは、印刷要求を出すときにバナーページの印刷を抑制するかどうかを選択できます。これは、lpadmin コマンドまたは Solaris プリンタマネージャから設定することができます。ユーザーが選択できるようにする場合、ユーザーがバナーページの印刷を抑制するには、`-o banner` オプションを使用する必要があります。

また、バナーページが不要な場合、バナーページの印刷を抑制してまったく印刷されないようにすることもできます。バナーページの印刷は、lpadmin コマンドを使用することによって抑制できます。

表 5-2 バナーページの印刷

使用するコマンド	バナーページの印刷	変更
lpadmin -p printer -o banner または lpadmin -p printer -o banner=always	常に行われる	一般ユーザーが lp -o nobanner コマンドを使用すると、要求は印刷されるが nobanner 引数は無視される root または別の権限のユーザーの場合、nobanner 引数を使用される
lpadmin -p printer -o nobanner lpadmin -p printer -o banner=optional	デフォルトで有効。ただし、lp -o nobanner コマンドを使えば要求単位で無効にできる	該当しない
lpadmin -p printer -o banner=never	無効	不可

詳細は、85 ページの「バナーページを抑制する方法」を参照してください。

▼ バナーページを選択可能にする方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. バナーページを選択可能にします。

```
# lpadmin -p printer-name -o banner=optional
```

`-p printer-name` バナーページ印刷を選択可能にするプリンタ名

`-o banner=optional` ユーザーが印刷要求を出すときにバナーページなしを指定できるようにする

すべての印刷要求でバナーページを強制印刷したい場合は、`-o banner=always` オプションを指定します。

バナーページの設定は、プリンタサーバーの `/etc/lp/printers/printer-name/configuration` ファイルに指定します。

3. 次のコマンドの出力には、「Banner not required」という行が入っています。

```
$ lpstat -p printer-name -l
```

例 — バナーページを選択可能にする

次の例は、プリンタ luna のバナーページを選択可能にする方法を示しています。

```
# lpadmin -p luna -o banner=optional
```

▼ バナーページを抑制する方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. バナー印刷を抑制します。

```
lpadmin -p printer-name -o banner=never
```

`-p printer-name` バナーページ印刷を選択可能にするプリンタ名

`-o banner=never` どのような状況でもバナーページ印刷を無効にする

バナーページの設定は、プリンタサーバーの `/etc/lp/printers/printer-name/configuration` ファイルに指定されます。

3. 次のコマンドの出力に `Banner not printed` という行が含まれていることを確認します。

```
$ lpstat -p printer-name -l
```

4. プリンタに印刷要求を送ってバナーページが印刷されないことを確認します。

例 — バナーページ印刷を抑制する

次の例は、プリンタ luna のバナーページを印刷しないようにする方法を示しています。

```
# lpadmin -p luna -o banner=never
```

プリンタクラスの設定

印刷サービスを使用すると、複数のローカルプリンタを1つのクラスにグループ化できます。この作業は、lpadmin -c コマンドを使用しなければ実行できません。

プリンタクラスを設定すると、ユーザーは印刷要求の出力先として(個々のプリンタではなく) そのクラスを指定できます。そのクラスで空いている最初のプリンタが印刷に使用されます。その結果、プリンタはできる限りビジーに保たれるので、応答時間が短縮されます。

印刷サービスに認識されるデフォルトのプリンタクラスはなく、定義したプリンタクラスのみが存在することになります。プリンタクラスを定義するには、次の3つの方法があります。

- プリンタタイプ別。たとえば、PostScript プリンタ
- 場所別。たとえば、5階に設置されたプリンタ
- 作業グループまたは部署別。たとえば、経理部

また、1つのクラスには特定の順序で使用される複数のプリンタを含めることができます。LP印刷サービスでは、常に各プリンタがクラスに追加された順番に従って利用できるプリンタをチェックします。したがって、最初に高速プリンタにアクセスした場合は、高速プリンタを低速プリンタよりも先にクラスに追加します。その結果、高速プリンタで最大限の印刷要求が処理されることとなります。低速プリンタは、高速プリンタが使用されているときのバックアッププリンタとして確保されます。

注 – 印刷要求の負荷は、ローカルプリンタのクラス内のプリンタ間でのみ調整されます。

クラス名も、プリンタ名と同様に固有の名前でなければなりません。クラス名は14文字以内の英数字で、下線を使用できます。

プリンタクラスは定義しなくてもかまいません。プリンタクラスを使用するとネットワーク上のユーザーに利点があると判断した場合にのみ、クラスを追加してください。

▼ プリンタのクラスを定義する方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. プリンタのクラスを定義します。

```
# lpadmin -p printer-name -c printer-class
```

-p *printer-name* プリンタのクラスに追加するプリンタ名

-c *printer-class* プリンタのクラス名

指定したプリンタが、プリンタサーバーの `/etc/lp/classes/printer-class` ファイル内でそのクラスのリストの最後に追加されます。プリンタクラスが存在しない場合は、作成されます。

3. プリンタがプリンタクラスの中にあることを確認します。

```
$ lpstat -c printer-class
```

例 — プリンタのクラスを定義する

次の例は、プリンタ luna をプリンタクラス roughdrafts に追加する方法を示しています。

```
# lpadmin -p luna -c roughdrafts
```

障害の通知の設定

事前に選択しておく、印刷サービスはプリンタ障害を検出したときに通知できます。プリンタの障害通知を受け取る方法として、以下のいずれかの方法を、lpadmin -A コマンドまたは Solaris プリンタマネージャを使用して選択することができます。

- root がログインしている端末にメッセージを書き込む
- root に電子メールを送る
- 通知しない

ただし、lpadmin -A コマンドを使用すると、選択したプログラムで指定されるメッセージを受信するようにすることもできます。また、lpadmin -A コマンドで、すでに知っているエラーに関する通知を選択的に抑制することもできます。

障害通知を配信するプログラムを指定しなければ、障害警告の内容は事前に定義済みのメッセージになります。このメッセージは、プリンタが印刷を停止しており、解決が必要であることを示します。

次の表は、lpadmin -A コマンドでプリンタに設定できる警告値を示しています。これらの警告値は、印字ホイール、フロントカートリッジ、フォームについても設定できます。

表 5-3 印刷障害の警告値

-A alert の値	説明
'mail [user-name]'	警告メッセージをプリンタサーバー上の root か lp、またはユーザー名として指定した user-name に電子メールで送信する
'write [user-name]'	警告メッセージをプリンタサーバー上の root か lp のコンソールウィンドウ、またはユーザー名として指定した user-name のコンソールウィンドウに送信する。指定したユーザーが警告メッセージを受け取るには、プリンタサーバーにログインしていなければならない
'command'	警告ごとに command ファイルを実行する。環境変数とカレントディレクトリは保存され、ファイルの実行時に復元される
quiet	障害が解決されるまで警告を停止する。この値は、ユーザー (root または指定したユーザー) が繰り返し警告を受け取るときに使用する
none	警告を送信しない。プリンタの障害警告を指定しない場合は、これがデフォルト値である

▼ プリンタの障害警告を設定する方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. プリンタに障害警告を設定します。

```
# lpadmin -p printer-name -A alert [-W minutes]
```

-p printer-name	プリンタ障害の警告を指定するプリンタ名
-A alert	プリンタ障害が起きたときに出力される警告の種類を指定する。alert の有効値については表 5-3 を参照してください。有効な値は mail、write、quiet など。
-W minutes	障害警告が出力される間隔 (分単位) を指定する。このオプションを指定しなければ、警告は一度だけ送信される

障害警告の設定は、プリンタサーバーの /etc/lp/printers/printer-name/alert.sh ファイルに入力されます。

3. 次のコマンドの出力から、「On fault」見出しに続く情報をチェックします。

```
$ lpstat -p printer-name -l
```

例 — プリンタの障害警告を設定する

次の例は、障害警告をユーザー joe に電子メールで送信し、その後は5分ごとに送信するプリンタ mars の設定方法を示しています。

```
# lpadmin -p mars -A 'mail joe' -W 5
```

次の例は、障害警告をコンソールウィンドウに送信し、その後は10分ごとに送信するプリンタ venus の設定方法を示しています。

```
# lpadmin -p venus -A write -W 10
```

次の例は、プリンタ mercury の障害警告を停止する方法を示しています。

```
# lpadmin -p mercury -A none
```

次の例は、プリンタ venus の障害が解決するまで、障害警告を停止する方法を示しています。

```
# lpadmin -p venus -A quiet
```

プリンタの障害回復の設定

障害通知を送信しないことを選択した場合には、問題を解決するために印刷障害を検出することができます。LP 印刷サービスは、障害のあるプリンタを継続して使用しません。プリンタ障害の警告に加えて、印刷要求が必要とするときに、印字ホイール、フォントカートリッジ、およびフォームを取り付けるようにシステム管理者に知らせる警告も設定できます。

lpadmin -F コマンドを使用すると、プリンタ専用の障害回復オプションを定義できます。これは、Solaris プリンタマネージャではできません。

プリンタ障害は、用紙切れやトナーカートリッジの交換が必要であるなど、きわめて単純な場合があります。より重大な問題としては、完全なプリンタ障害や電源障害などがあります。プリンタ障害を解決すると、障害が発生したときに有効だった印刷要求は、次のいずれかの方法で印刷を開始します。

- 印刷を最初から開始する
- 印刷を停止したページの先頭から印刷を再開する
- プリンタを使用可能にした後に、印刷を停止したページの先頭から印刷を再開する

印刷を停止したページの先頭から印刷を再開するには、印刷フィルタが必要です。印刷フィルタは、プリンタに使用される制御シーケンスを記録してページ境界を追跡します。この処理は、印刷サービスに使用されるデフォルトフィルタでは実行できません。指定した印刷フィルタで回復処理を実行できなければ、印刷サービスから通知されます。フィルタの作成方法については、153 ページの「新しい印刷フィルタを作成する方法」を参照してください。

プリンタ障害を解決した直後に印刷を再開したい場合は、`enable` コマンドを使用してプリンタを使用可能にします。

次の表は、`lpadmin -F` コマンドでプリンタに設定できる障害回復値を示しています。

表 5-4 プリンタ障害回復の値

<code>-F recover-options</code> の値	説明
<code>beginning</code>	障害回復後に、ファイルの先頭から印刷を再開する
<code>continue</code>	障害回復後に、印刷が停止されたページの先頭から印刷を開始する。この回復オプションには印刷フィルタが必要
<code>wait</code>	障害回復後に、プリンタを使用可能にするまで印刷が停止される。 <code>enable</code> コマンドでプリンタを使用可能になると、印刷は停止されたページの先頭から始まる。この回復オプションには印刷フィルタが必要

▼ プリンタの障害回復を設定する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. プリンタの障害回復を設定します。

```
# lpadmin -p printer-name -F recovery-options
```

<code>-p printer-name</code>	障害からの回復方法を指定するプリンタ名
<code>-F recovery-options</code>	<code>beginning</code> 、 <code>continue</code> 、 <code>wait</code> の 3 つの有効な回復オプション。 <code>recovery-options</code> の有効値については、表 5-4 を参照してください。

詳細については、`lpadmin(1M)` のマニュアルページを参照してください。障害回復の設定がプリンタサーバーの `/etc/lp/printers/printer-name/configuration` ファイルに入力されます。

3. 次のコマンドの出力から、「After fault」見出しに続く情報をチェックします。

```
$ lpstat -p printer-name -l
```

例 — プリンタの障害回復を設定する

次の例は、印刷が停止したページの先頭から再開させるプリンタ luna の設定方法を示しています。

```
# lpadmin -p luna -F continue
```

プリンタへのユーザーアクセスを制限する

利用できるプリンタの一部またはすべてにアクセスできるユーザーを制限する必要がある場合があります。たとえば、一部のユーザーが高品質プリンタ上で印刷できないようにして経費を抑えることができます。プリンタへのユーザーアクセスを制限するには、プリンタサーバー上でlpadmin -u コマンドを使用して「許可」リストと「拒否」リストを作成できます。Solaris プリンタマネージャを使用すると、許可リストのみを作成できます。どちらのリストも作成しなければ、プリンタはそこにアクセスできる全ユーザーが利用できます。

許可リストには、指定したプリンタへのアクセスを許可されるユーザー名が入っています。拒否リストには、指定したプリンタへのアクセスを拒否されるユーザー名が入っています。

許可リストと拒否リストには、次の規則が適用されます。

許可リストと拒否リストの規則	ユーザーアクセスの制限
許可リストも拒否リストも作成しない、または両方のリストが空	そのプリンタには全ユーザーがアクセスできる
許可リストで a11 を指定する	そのプリンタには全ユーザーがアクセスできる
拒否リストで a11 を指定する	サーバー上の root と lp 以外の全ユーザーのアクセスが拒否される
許可リストにエントリを作成する	拒否リストは無視される。リストに指定されているユーザーだけがプリンタにアクセスできる
拒否リストを作成し、許可リストは作成しないか許可リストを空にする	拒否リストで指定されたユーザーはプリンタにアクセスできない

実際にプリンタへのアクセスを制御しているのはプリンタサーバーなので、許可リストと拒否リストを作成できるのはプリンタサーバー上でだけです。許可リストと拒否リストを作成した場合、プリンタサーバーは、プリンタへのユーザーアクセスを排他的に制御します。

表 5-5 は、プリンタへのユーザーアクセスを制限するために許可リストまたは拒否リストに追加できる値を示しています。

表 5-5 許可リストと拒否リストの値

<i>user-list</i> の値	説明
<i>user</i>	任意のシステム上の特定ユーザー
<i>all</i>	すべてのシステム上の全ユーザー
<i>none</i>	すべてのシステム上の全ユーザーが該当しない
<i>system !user</i>	特定システム上の特定ユーザー
<i>!user</i>	ローカルシステム上の特定ユーザー
<i>all! user</i>	任意のシステム上の特定ユーザー
<i>all!all</i>	すべてのシステム上の全ユーザー
<i>system!all</i>	特定システム上の全ユーザー
<i>!all</i>	ローカルシステム上の全ユーザー

▼ プリンタへのユーザーアクセスを制限する方法

1. プリンタサーバーにスーパーユーザーまたは *lp* としてログインするか、同等の役割になります。
2. プリンタへのユーザーアクセスを許可または拒否します。

```
# lpadmin -p printer-name -u allow:user-list [deny:user-list]
```

<i>-p printer-name</i>	許可または拒否ユーザーアクセスリストを適用するプリンタ名
<i>-u allow: user-list</i>	許可ユーザーアクセスリストに追加するユーザー名。このコマンドで複数のユーザーを指定できる。空白またはコンマを使用して名前を区切る。空白を使用する場合は、名前のリストを引用符で囲む。 <i>user-list</i> の有効な値については、表 5-5 を参照
<i>-u deny: user-list</i>	拒否ユーザーアクセスリストに追加するユーザー名。このコマンドで複数のユーザーを指定できる。空白またはコンマを使用して名前を区切る。空白を使用する場合は、名前のリストを引用符で囲む。 <i>user-list</i> の有効な値については、表 5-5 を参照

指定したユーザーが、プリンタサーバーの次のファイル内でプリンタの許可または拒否リストに追加されます。

```
/etc/lp/printers/printer-name/users.allow  
/etc/lp/printers/printer-name/users.deny
```

注 – 許可リストの *user-list* に *none* を指定した場合、プリンタサーバー用に次のファイルは作成されません。

```
/etc/lp/printers/printer-name /alert.sh  
/etc/lp/printers/printer-name/alert.var  
/etc/lp/printers/printer-name/users.allow  
/etc/lp/printers/printer-name/users.deny
```

3. 次のコマンドの出力から、「Users allowed」または「Users denied」見出しに続く情報をチェックします。

```
$ lpstat -p printer-name -l
```

例 — プリンタへのユーザーアクセスを制限する

次の例は、nathan と george にだけプリンタ luna へのアクセスを許可する方法を示しています。

```
# lpadmin -p luna -u allow:nathan,george
```

次の例は、nathan と george にだけプリンタ asteroid へのアクセスを拒否する方法を示しています。

```
# lpadmin -p asteroid -u deny:"nathan george"
```

印刷要求の管理

ユーザーが印刷クライアントから印刷要求を出すと、その印刷要求はプリンタサーバー上の待ち行列に追加されてからプリンタに送信されます。印刷要求が待ち行列に入っている間は、クライアントシステムからの要求で取り消したり、その状態情報を取得したりできます。LP コマンドで印刷要求の移動、停止、再開または優先順位の変更などを実行するには、プリンタサーバーにログインする必要があります。これらの動作によって、印刷サービスを絶えずスムーズに機能させることができます。

次の表は、lp -H コマンドを使用して印刷要求の優先順位を変更する値を示しています。

表 5-6 印刷要求の優先順位を変更する値

-H <i>change-priority</i> の値	説明
hold	ユーザーが取り消すか、LP 印刷サービスに要求の印刷再開を指示するまで、印刷要求を停止する
resume	停止されていた印刷要求を待ち行列に戻す。印刷要求は、優先順位と待ち行列内での位置に従って印刷される。すでに印刷中の印刷ジョブを停止すると、resume は印刷要求が次に印刷される要求になるように待ち行列の先頭に配置する
immediate	印刷要求を待ち行列の先頭に配置する。要求がすでに印刷中であれば、次の要求をすぐに印刷できるように、印刷中の要求を停止する

▼ 印刷要求の状態をチェックする方法

1. ネットワーク上の任意のシステムにログインします。
2. プリンタと印刷要求の状態をチェックします。

ここでは、最も一般的に使用するオプションだけを掲載してあります。その他の有効なオプションについては、lpstat (1) のマニュアルページを参照してください。

```
$ lpstat -o [list] | -u [user-list]
```

-o <i>list</i>	<p>特定のプリンタの印刷要求の状態が表示される。<i>list</i> には、1 つ以上のプリンタ名、プリンタクラス名、または印刷要求 ID を指定できる。</p> <p><i>list</i> には、複数のプリンタ名、クラス名、ID を指定できる。値を区切るには空白またはコンマを使用する。空白を使用する場合は、値のリストを引用符で囲む。</p> <p><i>list</i> を指定しなければ、すべてのプリンタへの印刷要求の状態が表示される</p>
-u <i>user-list</i>	<p>特定のユーザーの印刷要求の状態が表示される。<i>user-list</i> では複数のユーザー名を指定できる。</p> <p>このコマンドで複数のユーザーを指定できる。ユーザー名を区切るには空白またはコンマを使用する。空白を使用する場合は、名前のリストを引用符で囲む。</p> <p><i>user-list</i> を指定しなければ、すべてのユーザーの印刷要求の状態が表示される</p>

lpstat コマンドを使用して印刷要求の状態をチェックすると、印刷要求ごとに 1 行ずつ表示されます。各行には、左から右に要求 ID、ユーザー、出力バイト数、要求日時、「being filtered」のような要求に関する情報が表示されます。

例 — 印刷要求の状態をチェックする

次の例は、ユーザー fred にはプリンタ luna への印刷要求が 1 つあることを示しています。

```
$ lpstat
luna-1    fred      1261     Jul 12 17:34
```

次の例は、ユーザー paul には待ち行列に入っている印刷要求がないことを示しています。

```
$ lpstat -u paul
```

次の例は、プリンタ moon の待ち行列に入っている印刷要求が 2 つあることを示しています。

```
$ lpstat -o moon
moon-78   root      1024     Jul 14 09:07
moon-79   root      1024     Jul 14 09:08
```

プリンタの印刷の有効化と無効化

enable コマンドと disable コマンドは、プリンタが待ち行列に入っている印刷要求を印刷するか保留するかを制御します。プリンタを使用不可にすると、プリンタは待ち行列内の要求の印刷を保留します。ただし、要求はそのまま待ち行列に追加されたままです。要求が待ち行列に追加されないようにするには、プリンタを設定して印刷要求を拒否させなければなりません。印刷要求の拒否については、97 ページの「印刷要求の受け付けまたは拒否」を参照してください。

Solaris プリンタマネージャを使用してプリンタを追加すると、プリンタは有効になり印刷要求を受け付けます。Solaris プリンタマネージャは、それ以上のプリンタ管理は提供しません。

プリンタが使用不可になっている場合は、使用可能にしなければなりません。この状態は、プリンタ障害が起きると発生することがあります。プリンタを使用可能にすると、印刷サービスがそれ以後に印刷待ち行列の要求を拒否しても、待ち行列が空になるまで、印刷待ち行列からの要求が印刷されます。

次の図は、プリンタが使用不可になったときに印刷要求の処理が中断される様子を示しています。

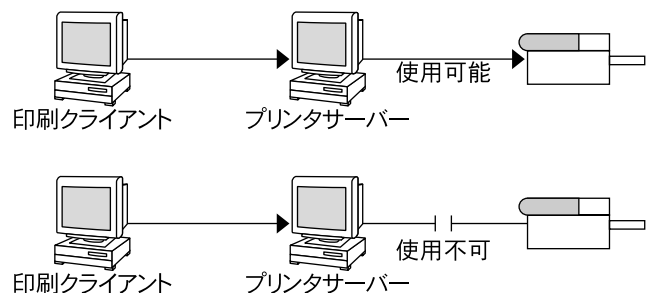


図 5-1 プリンタが使用可能または使用不可になる場合

▼ プリンタを使用可能または使用不可にする方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. 印刷要求の印刷を停止します。

```
# disable [-c | -W] [-r "reason"] printer-name
```

<code>disable</code>	現在のジョブを取り消してから、プリンタを使用不可にする。現在のジョブは保存され、プリンタが使用可能になると、印刷が再開される。
<code>-c</code>	現在のジョブを取り消してから、プリンタを使用不可にする。現在のジョブが後から印刷されることはない
<code>-W</code>	現在のジョブが終了するまで待ってからプリンタを使用不可にする
<code>-r "reason"</code>	プリンタが使用不可になった理由をユーザーに示す。理由は保存され、ユーザーがプリンタの状態を確認すると (lpstat -p コマンド)、表示される
<code>printer-name</code>	印刷要求の印刷を停止するプリンタ名

注 – プリンタのクラスを使用可能または使用不可にすることはできません。個々のプリンタのみを使用可能または使用不可にすることができます。

3. 印刷要求の印刷を開始します。

```
# enable printer-name
```

4. プリンタが使用可能になっていることを確認します。


```
$ lpstat -p printer-name
```

例 — プリンタを使用可能または使用不可にする

次の例は、プリンタ luna 上の現在のジョブを停止し、後から印刷できるように保存して、プリンタが印刷要求の印刷を停止した理由を表示する方法を示しています。

```
# disable -r "changing the form" luna
```

次の例は、プリンタ luna の印刷要求の印刷を開始する方法を示しています。

```
# enable luna
printer "luna" enabled
```

印刷要求の受け付けまたは拒否

accept コマンドと reject コマンドを使用すると、印刷要求が格納される印刷待ち行列のオンとオフを切り替えることができます。

reject コマンドを使用すると、プリンタの印刷待ち行列がオフになります。新しい印刷要求はプリンタサーバーの待ち行列に入れません。その待ち行列に入っているすべての印刷要求は、そのまま印刷されます。すでに待ち行列に入っている要求の印刷を停止したい場合は、そのプリンタを使用不可にしなければなりません。次の表は、accept、reject、enable、disable の各コマンドの機能を比較したものです。

表 5-7 accept、reject、enable、disable の各コマンドの機能

コマンド	機能
accept	印刷待ち行列に送信された印刷要求を受け付ける
enable	印刷待ち行列にある要求を印刷する
reject	印刷待ち行列に送信された印刷要求を拒否する
disable	現在印刷待ち行列にある印刷要求を停止する

印刷要求が拒否されると、印刷サービスはその要求を出したユーザーにメッセージを送り、指定されたプリンタには印刷要求が受け付けられていないことを通知します。

また、要求を受け付けない理由をコマンド行から指定できます。その理由は、ユーザーがプリンタの待ち行列をチェックしようとするユーザーのシステムに表示されます。次の図は、印刷待ち行列が拒否されたときに印刷要求が中断される様子を示しています。

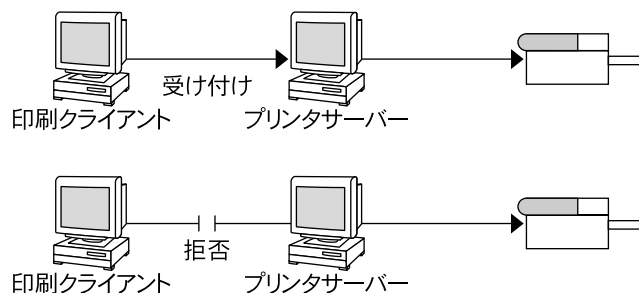


図 5-2 印刷待ち行列が要求を受け付けるか拒否する場合

▼ プリンタへの印刷要求を受け付けるまたは拒否する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. プリンタの印刷要求を停止します。

```
# reject [-r "reason"] printer-name
```

`-r "reason"` プリンタが印刷要求を拒否している理由をユーザーに示す。理由は格納され、ユーザーがプリンタの状態をチェックする (`lpstat -p` コマンド) と表示される

`printer-name` 印刷要求を停止するプリンタ名

待ち行列に入れられた要求は、プリンタが使用可能になっている限りそのまま印刷されます。印刷を停止するようにプリンタを使用不可にする手順については、96 ページの「プリンタを使用可能または使用不可にする方法」を参照してください。

3. プリンタの印刷要求を開始します。

```
# accept printer-name
```

4. プリンタの状態が、印刷要求を受け付けているか、拒否しているかを確認します。

```
$ lpstat -p printer-name
```

例 — プリンタへの印刷要求を受け付けるまたは拒否する

次の例は、プリンタ `luna` の印刷要求の受け入れを停止する方法を示しています。

```
# reject -r "luna is down for repairs" luna
destination "luna" will no longer accept requests
```

次の例は、プリンタ luna が印刷要求を受け入れるように設定する方法を示しています。

```
# accept luna
destination "luna" now accepting requests
```

印刷要求の取り消し

cancel コマンドを使用すると、印刷待ち行列からの印刷要求を取り消したり、印刷中のジョブを取り消すことができます。cancel コマンドには、次の3つの使用方法があります。

- 要求識別番号 (要求 ID) を使用して要求を取り消す。
- すべてまたは指定したプリンタ上で特定のユーザーからの要求を取り消す。
- 現在印刷中のジョブを取り消す。

cancel コマンドを使用すると、要求が取り消され、待ち行列内の次の要求が印刷されることを示すメッセージが表示されます。次の場合にのみ、印刷要求を取り消すことができます。

- 要求を出したユーザーが、要求を出したシステムにログインしている状態で取り消しを実行した場合
- プリンタサーバーの /etc/printers.conf ファイルで user-equivalence オプションが構成されていて、要求を出したユーザーが任意のクライアントシステムから取り消しを実行した場合
- プリンタサーバーにスーパーユーザーまたは lp としてログインしているか、同等の役割になっている場合

特定の要求を取り消すには、その要求 ID を知る必要があります。リクエスト ID は必ず、プリンタ名、ハイフン、印刷要求番号から構成されています。たとえば、luna-185 となります。

印刷要求を依頼すると、その要求 ID が表示されます。印刷要求 ID を忘れた場合は、-o printer オプションを指定して lpstat コマンドを使用すると ID を調べることができます。

▼ 印刷要求を取り消す方法

1. 他のユーザーの印刷要求を取り消すには、スーパーユーザーまたは lp になるか、同等の役割になります。
2. 取り消す印刷要求の要求 ID を決めます。

```
# lpstat
```

詳細は、94 ページの「印刷要求の状態をチェックする方法」を参照してください。

3. 印刷要求を取り消します。

```
$ cancel request-id | printer-name
```

<i>request-id</i>	取り消す印刷要求の要求 ID。このコマンドで複数の要求 ID を指定できる。空白またはコンマを使用する。空白を使用する場合は、要求 ID のリストを引用符で囲む
<i>printer-name</i>	現在印刷中の印刷要求を取り消したいプリンタを指定する。 このコマンドで複数のプリンタ名を指定できる。プリンタ名を区切るには空白またはコンマを使用する。空白を使用する場合は、印刷クライアントのリストを引用符で囲む。

4. 印刷要求が取り消されていることを確認します。

```
$ lpstat -o printer-name
```

例 — 印刷要求を取り消す

次の例は、luna-3 と luna-4 の印刷要求を取り消す方法を示しています。

```
$ cancel luna-3 luna-4  
request "luna-3" cancelled  
request "luna-4" cancelled
```

次の例は、プリンタ luna 上で現在印刷中の印刷要求を取り消す方法を示しています。

```
# cancel luna  
request "luna-9" cancelled
```

▼ 特定のユーザーからの印刷要求を取り消す方法

1. (省略可能) 他のユーザーの印刷要求を取り消すには、スーパーユーザーまたは lp になるか、同等の役割になります。
2. 特定のユーザーから印刷要求を取り消します。

```
$ cancel -u user-list [printer-name]
```

<i>-u user-list</i>	特定のユーザーの印刷要求を取り消す。 <i>user-list</i> では複数のユーザー名を指定できる。ユーザー名を区切るには空白またはコンマを使用する。空白を使用する場合は、名前のリストを引用符で囲む。
---------------------	--

printer-name 指定したユーザーの印刷要求を取り消したいプリンタを指定する

printer-name では複数のプリンタ名を指定できる。プリンタ名を区切るには空白またはコンマを使用する。空白を使用する場合は、プリンタ名のリストを引用符で囲む

*printer-name*を指定しないと、ユーザーの印刷要求はすべてのプリンタで取り消される

例 — 特定のユーザーからの印刷要求を取り消す

次の例は、プリンタ luna 上でユーザー george から依頼されたすべての印刷要求を取り消す方法を示しています。

```
# cancel -u george luna
request "luna-23" cancelled
```

次の例は、ユーザー george から依頼されたすべての印刷要求をすべてのプリンタ上で取り消す方法を示しています。

```
# cancel -u george
request "asteroid-3" cancelled
request "luna-8" cancelled
```

印刷要求の移動

プリンタの使用方法を変更する計画がある場合や、プリンタの使用を中止する場合は、LP 印刷サービスを設定して新たな印刷要求を拒否しなければなりません。さらに、現在印刷待ち行列に入っている要求があれば、移動するか取り消すかする必要があります。lpmove コマンドを使用すると、個々の印刷要求またはすべての印刷要求を別のローカルプリンタに移動できます。

要求 ID は印刷要求を移動しても変更されないため、ユーザーは引き続き各自の要求を調べることができます。新しく指定したプリンタでは満たせない要件 (ファイル内容形式やフォームなど) を持つ印刷要求は移動できません。この種の印刷要求は取り消さなければなりません。

▼ 印刷要求を別のプリンタに移動する方法

あるプリンタから別のプリンタにすべての要求を移動する場合は、要求 ID がわからなくてもかまいません。ただし移動する前に、影響を受ける印刷要求の数を調べておくといでしょう。

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. (省略可能) 元のプリンタ上で印刷要求の要求 ID をチェックします。

```
# lpstat -o printer-name1
```

3. (省略可能) 宛先プリンタが印刷要求を受け付けているかどうかをチェックします。

```
# lpstat -p printer-name2
```

```
-p printer-name2          印刷要求の移動先となるプリンタ名
```

4. 元のプリンタから宛先プリンタにすべての印刷要求を移動します。

```
# lpmove printer-name1 printer-name2
```

```
printer-name1          すべての印刷要求の移動元となるプリンタ名  
printer-name2          すべての印刷要求の移動先となるプリンタ名
```

詳細は、lpmove (1M) のマニュアルページを参照してください。

一部の要求を宛先プリンタ上で印刷できない場合は、元のプリンタの待ち行列内に残ります。要求 ID を使用すると、lpmove コマンドで特定の印刷要求を別のプリンタに移動することもできます。

5. 元のプリンタ上で印刷要求の受け付けを開始します。

すべての印刷要求を別のプリンタに移動すると、lpmove コマンドはそのプリンタへの印刷要求の受け付けを自動的に停止します。そのプリンタへの新しい印刷要求の受け付けを開始したい場合は、この手順が必要です。

```
# accept printer-name1
```

6. 移動元のプリンタの待ち行列に残っている印刷要求をチェックします。

```
$ lpstat -o printer-name1
```

次のコマンドを使用して、すべての指定した印刷要求が宛先のプリンタ待ち行列に移動していることを確認します。

```
$ lpstat -o printer-name2
```

例 — 印刷要求を別のプリンタに移動する

次の例は、プリンタ luna からプリンタ terra に印刷要求を移動し、移動元プリンタ luna に印刷要求の受付を再開させる方法を示しています。

```
# lpmove luna terra  
# accept luna
```

印刷要求の優先順位の変更

印刷要求を出し終わったら、その優先順位をプリンタサーバーの待ち行列内で次のように変更できます。

- 印刷が終了していない場合は、その印刷要求を停止できます。要求を保留状態にして停止すると、現在印刷されている場合は、再開するまで印刷されません。他の印刷要求は、停止中の要求よりも先に印刷されます。
- 任意の印刷要求を待ち行列の先頭に移動できます。その場合、先頭の印刷要求は次に印刷されます。ジョブの印刷を即座に開始したい場合は、現在印刷中のジョブを停止して中断できます。
- 引き続き印刷したいジョブの優先順位を変更して、待ち行列内で優先順位が低い要求の前と、優先順位が同じか高い要求の後の間に移動できます。

▼ 印刷要求の優先順位を変更する方法

1. 印刷要求を保持するプリンタサーバーに、スーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. 優先順位を変更したい印刷要求の要求 ID を決めます。

```
# lpstat
```

詳細は、94 ページの「印刷要求の状態をチェックする方法」を参照してください。

3. 印刷要求の優先順位を変更します。

```
# lp -i request-id -H change-priority
```

```
-i request-id
```

変更したい印刷要求の要求 ID。

このコマンドで複数の要求 ID を指定できる。空白またはコンマを使用する。空白を使用する場合は、要求 ID のリストを引用符で囲む

```
-H change-priority
```

印刷要求の優先順位を変更する方法として、hold、resume、immediate のいずれかを指定する。

change-priority の有効値については、表 5-6 を参照してください。

また、lp の -q コマンドを使用すると、指定した印刷要求の優先順位を変更できます。優先順位は、最上位の 0 から最下位の 39 までの間で変更できます。

例 — 印刷要求の優先順位を変更する

次の例は、要求 ID が asteroid-79 の印刷要求を優先順位 1 に変更する方法を示しています。

```
# lp -i asteroid-79 -q 1
```


第 6 章

文字セット、フィルタ、フォーム、 フォントの管理 (手順)

この章では、文字セット、印刷フィルタ、フォーム、およびフォントについての基本事項と、それらを設定して管理する手順について説明します。

この章で説明する手順は次のとおりです。

- 108 ページの「印字ホイールとフォントカートリッジを定義する方法」
- 109 ページの「印字ホイールまたはフォントカートリッジを取り外すまたは装着する方法」
- 111 ページの「印字ホイールまたはフォントカートリッジの装着を促す警告を設定する方法」
- 112 ページの「選択可能文字セットの別名を設定する方法」
- 116 ページの「印刷フィルタを追加する方法」
- 116 ページの「印刷フィルタを削除する方法」
- 117 ページの「印刷フィルタに関する情報を表示する方法」
- 120 ページの「フォームを追加する方法」
- 121 ページの「フォームを削除する方法」
- 122 ページの「フォームを取り外し、装着する方法」
- 123 ページの「フォームの装着に関する警告を設定する方法」
- 125 ページの「フォームに関する情報を表示する方法」
- 125 ページの「フォームの現在の状態を表示する方法」
- 126 ページの「フォームへのユーザーアクセスを制限する方法」
- 127 ページの「フォームへのプリンタアクセスを制限する方法」
- 130 ページの「ダウンロードされた PostScript フォントをインストールする方法」
- 131 ページの「ホスト常駐 PostScript フォントをインストールする方法」

印刷については、第 2 章を参照してください。

文字セットの管理

プリンタでテキストを各種フォント書体で印刷する方法は、それぞれ異なります。たとえば、PostScript プリンタは、テキストをグラフィックスとして処理します。これらのプリンタは、複数のフォントを使用してテキストを生成し、ページ上の任意の位置、サイズ、または方向にテキストを配置できます。その他の形式のプリンタは、印字ホイール、フォントカートリッジ、プログラムされた選択可能な文字セットのいずれかを使用するため、フォントの種類と大きさには制限があります。通常、1つのプリンタ形式には1つの印刷方法が適用されます。

必要に応じてプリンタにフォントを装着する必要があるという点で、LP 印刷サービスでは印字ホイールとフォントカートリッジを同様に扱うことができます。ホイールまたはカートリッジを物理的に装着する必要がある文字セットを、「プリンタに装着する文字セット」といいます。物理的に装着する必要がなく、プリンタにあらかじめプログラムされていて、印刷要求によって選択可能な文字セットを、「プリンタに組み込みの文字セット」といいます。

PostScript 以外のプリンタを設定する場合は、ユーザーが利用可能な、印字ホイールまたは選択可能な文字セットを LP 印刷サービスに指定する必要があります。ユーザーが印刷要求を出すときに、`lp -s` コマンドを使用すると、ジョブの印刷に使用する印字ホイールまたは選択可能な文字セットを指定できます。ユーザーは、すでに定義してある名前でもフォントを参照するだけなので、実際に使用される文字セットの種類を知る必要はありません。たとえば、印刷ホイールを `gothic` と定義したとします。この `gothic` 印刷ホイールを要求するには、次のコマンドを使用します。

```
% lp -s gothic filename
```

選択可能な文字セット

プリンタによってサポートされる選択可能文字セットは、そのプリンタの `terminfo` エントリに表示されています。たとえば、`ln03` プリンタのエントリは、`/usr/share/lib/terminfo/1/ln03` です。`tput` コマンドを使用して、`terminfo` データベースの任意のプリンタタイプの選択可能文字セットの名前を選択できます。`tput` コマンドの構文は次のとおりです。

```
tput -T printer-type csn
```

`csn` オプションは文字セット番号 (character set number) の省略形です。番号は、プリンタが初期化された後に常に設定されるデフォルトの文字セット番号である 0 で始まります。その他の文字セット名を表示するには、-0 の代わりに -1、-2、-3 などを使用してコマンドを繰り返してください。選択可能文字セットごとに、`terminfo` 名 (たとえば `usascii`、`english`、`finnish` など) が返されます。

通常、`terminfo` 文字セット名は、プリンタのマニュアルで使用されている文字セット名となるべく一致させてください。同じ文字セット名を使用しないメーカーもあるため、`terminfo` 名はプリンタタイプごとに異なる場合があります。

LP 印刷サービスを使用して選択可能文字セットを登録する必要はありません。ただし、より意味のある名前または別名を与えることができます。

注 – プリンタで使用できる選択可能文字セットを指定しない場合、LP 印刷サービスは、プリンタが任意の文字セット名 (cs0、cs1、cs2 など) またはプリンタが認識する terminfo 名を受け付けることができると仮定します。

lpstat -p -l コマンドを使用して、プリンタサーバーに接続されているプリンタごとに、定義されている選択可能文字セット名を表示できます。

注 – PostScript のフォントは、terminfo データベースのエントリではなく PostScript フィルタによって制御されるため、lpstat -p -l コマンドを使用しても PostScript プリンタ用の文字セットは表示されません。PostScript フォントの管理方法については、128 ページの「フォントの管理」を参照してください。

プリンタに装着する文字セット

別の文字セットを使用するもう一つの方法は、物理的にプリンタに装着できる取り外し可能な印字ホイールまたはフォントカートリッジを使用することです。

プリンタに装着する文字セットを管理するには、LP 印刷サービスに、使用したい印字ホイール名と、プリンタが異なる印字ホイールを必要とするときの警告方法を指定します。次に、ユーザーが lp -s コマンドを使用して特定の文字セットを要求すると、スケジューラは印字ホイールを装着するよう警告を送信し、印刷要求が印刷待ち行列に入れられます。正しい印字ホイールを装着して、印字ホイールを装着したことを LP 印刷サービスに指示すると、ジョブが印刷されます。詳細は、109 ページの「印字ホイールまたはフォントカートリッジを取り外すまたは装着する方法」を参照してください。

1 台のプリンタに対して複数の印字ホイールやカートリッジを指定しなければ、LP 印刷サービスは、プリンタが 1 つの固定印字ホイールまたはカートリッジしか持っておらず、ユーザーはプリンタを使用する際に特殊な印字ホイールやカートリッジを指定できないと見なします。

選択可能文字セットとは違って、印字ホイールまたはカートリッジ用に選択する名前は、terminfo データベースのエントリとは関係がありません。印字ホイール名またはカートリッジ名は、ユーザーが LP 印刷サービスと通信を行うためにだけ使用されます。

ただし、印字ホイールまたはカートリッジ用に選択する名前は、ユーザーがわかりやすいものにしてください。その名前がフォントの書体を表すようにしてください。さらに、その名前は、同じ種類の印字ホイールやカートリッジ、または選択可能文字

セットを持つプリンタの場合には、同じ名前にします。それによって、ユーザーは、どのプリンタ、印字ホイール、カートリッジ、選択可能文字セットを使用するかに関係なく、フォントの書体 (文字セット) を指定できます。

システム管理者とプリンタユーザーは、印字ホイールまたはカートリッジに同じ名前を使用してください。そうしないと、ユーザーが指定する文字セットと管理者が装着するものが異なる可能性があります。

印字ホイールの確認

印字ホイールを確認する手順は、フォームを確認する手順と似ています。一部のプリンタは (通常、文字ベースの印字を行うプリンタ)、特定のフォントや文字セットを提供する印字ホイールや印字カートリッジのような、取り外し可能な印字ヘッドを持っています。ユーザーは名前の付いた文字セットを要求できます。その文字セットがない場合、LP 印刷サービスは要求元または管理者に通知します。印刷ジョブは、印字ホイールが変更されるまで、印刷待ち行列に格納されます。

印字ホイールまたはカートリッジの装着の警告

LP 印刷サービスから出す警告を指定するのと同じ方法で、印字ホイールまたはカートリッジを装着する際に出す警告を指定します。警告の概要については、87 ページの「障害の通知の設定」を参照してください。

▼ 印字ホイールとフォントカートリッジを定義する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. プリンタで使用できる印字ホイールまたはフォントカートリッジを定義します。

```
print-server# lpadmin -p printer-name -s hard-charset1[,hard-charset2...]
```

`-p printer-name`

プリンタで使用できる印字ホイールまたはフォントカートリッジを定義する

```
-s hard-charset
```

印字ホイールまたはフォントカートリッジを定義するプリンタ名。

このコマンドで複数のプリンタに装着する文字セット名を指定できる。文字セット名を区切るには空白またはコンマを使用する。空白を使用する場合は、文字セット名のリストを引用符で囲む。

ユーザーにとって意味のある名前を定義して、ユーザーに通知する

印字ホイールまたはフォントカートリッジの定義が、プリンタサーバーの `/etc/lp/printers/printer-name/configuration` ファイルに追加されます。

3. プリンタサーバーの印刷クライアントにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
4. 印刷クライアントに対して同じ印字ホイールまたはフォントカートリッジを定義します。

```
print-client# lpadmin -p printer-name -S hard-charset1[,hard-charset2...]
```

このコマンドの変数は、手順 2 と同じです。

印字ホイールまたはフォントカートリッジの定義が、印刷クライアントの `/etc/lp/printers/printer-name/configuration` ファイルに追加されます。

5. 印字ホイールまたはフォントカートリッジを使用する必要がある印刷クライアントごとに、手順 3 と手順 4 を繰り返します。
6. プリンタサーバーと印刷クライアント上で、次のコマンド出力の「Character sets」見出しの後にある情報を確認します。

```
$ lpstat -p printer-name -l
```

例 — 印字ホイールを定義する

次の例は、印刷クライアント `asteroid` のプリンタ `luna` 上で印字ホイール `pica` を定義する方法を示しています。

```
asteroid# lpadmin -p luna -S pica
```

▼ 印字ホイールまたはフォントカートリッジを取り外すまたは装着する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. プリンタ内の印字ホイールまたはフォントカートリッジを取り外します。

```
# lpadmin -p printer-name -M -S none
```

```
-p printer-name          印字ホイールまたはフォントカートリッジを取り外すプリンタ名
```

```
-M -S none                現在の印字ホイールまたはフォントカートリッジを取り外すように指定する
```

詳細については、`lpadmin(1M)` のマニュアルページを参照してください。
現在の印字ホイールまたはフォントカートリッジが、プリンタサーバーの
`/etc/lp/printers/printer-name/configuration` ファイルから削除されます。

3. 印字ホイールまたはフォントカートリッジをプリンタから削除します。
4. プリンタに新しい印字ホイールまたはフォントカートリッジを入れます。
5. 新しい印刷ホイールまたはフォントカートリッジを装着します。

```
# lpadmin -p printer-name -M -S hard-charset
```

```
-p printer-name          印字ホイールまたはフォントカートリッジを装着するプリンタ名
```

```
-M -S hard-charset      装着したい印字ホイールまたはフォントカートリッジのプリンタに装着する文字セット名
```

印字ホイールまたはフォントカートリッジが、プリンタサーバーの
`/etc/lp/printers/printer-name/configuration` ファイルに追加されます。装着された印字ホイールまたはフォントカートリッジは、取り外されるか、新しいものが装着されるまで使用可能です。

6. 次のコマンドの出力の中で、「Print wheels」または「Character set」の見出しの下にある情報をチェックします。印刷ホイール名または文字セット名と注意「mounted」が表示されます。

```
$ lpstat -p printer-name -l
```

例 — 印字ホイールを取り外すまたは装着する

次の例は、プリンタ `luna` の現在の印刷ホイールを取り外し、`pica` 印刷ホイールを装着する方法を示しています。

```
# lpadmin -p luna -M -S none
# lpadmin -p luna -M -S pica
```

▼ 印字ホイールまたはフォントカートリッジの装着を促す警告を設定する方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. 印刷ホイールまたはフォントカートリッジの装着を促す警告を設定します。

```
# lpadmin -S hard-charset -A alert [-Q requests] [-W minutes]
```

<code>-S hard-charset</code>	警告を設定したい印字ホイールまたはフォントカートリッジのプリンタに装着する文字セット名
<code>-A alert</code>	印字ホイールまたはフォントカートリッジが要求されたときに出力される警告の種類を指定する。 <code>alert</code> の有効値については、表 5-3 を参照してください。有効な値は <code>mail</code> 、 <code>write</code> 、 <code>quiet</code> など。 mail または write を指定すると、あらかじめ定義された警告メッセージが表示される。この警告メッセージは、指定した印字ホイールまたはフォントカートリッジの装着を促すもので、それを使用するように設定されている 1 つ以上のプリンタ名が含まれる
<code>-Q requests</code>	警告が出される前に、印字ホイールまたはフォントカートリッジが、待ち行列に入っていないなければならない印刷要求の数を指定する。このオプションを指定しなければ、待ち行列に印刷要求が 1 つ入っただけで警告が出される
<code>-W minutes</code>	警告が出される間隔 (分単位) を指定する。このオプションを指定しなければ、警告は一度だけ送られる

詳細については、`lpadmin(1M)` のマニュアルページを参照してください。

警告は、プリンタサーバーの `/etc/lp/pwheels/charset-name/alert.sh` ファイルに追加されます。

3. 次のコマンドの出力をチェックして、印字ホイールまたはフォントカートリッジの装着を促す警告が追加されているかどうかを確認します。

```
# lpadmin -S hard-charset -A list
```

あるいは、警告を出すために印刷要求に低い番号を設定した場合、最小限の要求を満たすために十分な印刷要求を出し、印字ホイールまたはフォントカートリッジの装着を促す警告を受け取ることを確認します。

例 — 印字ホイールまたはフォントカートリッジの装着を促す警告を設定する

次の例は、印刷待ち行列に elite 印字ホイールに対する 10 の印刷要求があるとき、elite に関して 5 分間隔で電子メールで警告が送られるように設定する方法を示しています。

```
# lpadmin -S elite -A mail -Q 10 -W 5
```

次の例は、印刷待ち行列に finnish フォントカートリッジに対する 5 つの印刷要求があるときに、finnish に関して 1 分間隔で電子メールで警告が送られるように設定する方法を示しています。

```
# lpadmin -S finnish -A mail -Q 5 -W 1
```

次の例は、印刷待ち行列に elite 印字ホイールに対する 5 つの印刷要求があるときに、elite に関して 10 分間隔でコンソールウィンドウに警告が送られるように設定する方法を示しています。

```
# lpadmin -S elite -A write -Q 5 -W 10
```

次の例は、elite 印字ホイールに警告が送られないように設定する方法を示します。

```
# lpadmin -S elite -A none
```

▼ 選択可能文字セットの別名を設定する方法

選択可能文字セットの terminfo 名が正しい場合は、この手順を実行する必要はありません。terminfo データベースの使用については、135 ページの「サポートされていないプリンタの terminfo エントリを追加する」を参照してください。

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. 指定したプリンタタイプの選択可能文字セット名を表示します。

```
# tput -T printer-type csn
```

-T printer-type	terminfo データベースに入っているプリンタタイプ。 terminfo データベースのエントリについては、48 ページの「プリンタタイプ」を参照
n	指定したプリンタタイプの選択可能文字セットを表す番号 (0、1、2、3、4、5 など)。プロンプト記号に続いて選択可能文字セット名が表示される。たとえば、cs1 と指定すると、english# と表示される

詳細は、tput (1) のマニュアルページを参照してください。

3. 選択可能文字セットの別名を設定します。

```
# lpadmin -p printer-name -s select-charset1=alias1[,select-charset2=alias2...]
```

<code>-p printer-name</code>	選択可能文字セットの別名を設定するプリンタ名
<code>-s select-charset</code>	別名を設定する選択可能文字セット名。この名前は、手順 2 で検索できる
<code>alias</code>	指定した選択可能文字セットの別名。選択可能文字セット名の他に、この別名を使用できる。 このコマンドで複数の別名を設定できる。別名を区切るには空白またはコンマを使用する。空白を使用する場合は、別名のリストを引用符で囲む

別名は、プリンタサーバーの `/etc/lp/printers/printer-name/configuration` ファイルに追加されます。

4. プリンタサーバーの印刷クライアントであるシステムにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
5. 選択可能文字セットの別名を設定します。

```
# lpadmin -p printer-name -s select-charset1=alias1[,select-charset2=alias2...]
```

このコマンドの変数は、手順 3 と同じです。

別名は印刷クライアントの `/etc/lp/printers/printer-name/configuration` ファイルに追加されます。

6. 別名を使用する必要がある印刷クライアントごとに、手順 4 と手順 5 を繰り返します。
7. プリンタサーバーと印刷クライアント上で、次のコマンドの出力の中に選択可能文字セットの別名のリストがあることを確認します。

```
$ lpstat -p printer-name -l
```

または、選択可能文字セットに別名を使用する印刷要求を出して、出力をチェックします。

例 — 選択可能文字セットの別名を設定する

次の例は、選択可能文字セット名を表示し、ln03 プリンタタイプのプリンタ luna 上の usascii 選択可能文字セットの別名として text を指定する方法を示しています。

```
# tput -T ln03 cs0
usascii# tput -T ln03 cs1
english# tput -T ln03 csn2
```

```
finnish# tput -T ln03 csn3
japanese# tput -T ln03 cs4
norwegian#
# lpadmin -p luna -S usascii=text
```

印刷フィルタの管理

「印刷フィルタ」とは、ファイルの内容形式を出力先プリンタが受け付けられる内容形式に変換するプログラムのことです。LP 印刷サービスはフィルタを使用して、次の機能を提供します。

- ファイルを特定タイプのプリンタで正しく印刷できるように、1つのデータ形式から別のデータ形式に変換する。
- 両面印刷、横方向印刷、ドラフト印刷、または高品質印刷などの特別な印刷モードを処理する。
- プリンタの障害を検出して LP 印刷サービスに通知する。その結果、印刷サービスはユーザーとシステム管理者に警告を出すことができる。

すべての印刷フィルタが上記のすべての機能を実行できるわけではありません。各機能はプリンタに固有なので、別々に実装できます。

LP 印刷サービスは、表 6-1 に示す PostScript フィルタを提供します。これらのフィルタプログラムは、`/usr/lib/lp/postscript` ディレクトリに入っています。通常、PostScript 印刷を行う場合は、プリンタサーバーの設定時にフィルタプログラムをインストールする以外に何も行う必要はありません。Solaris プリンタマネージャが提供されるフィルタを自動的に使用可能にします。ただし、他のプリンタを管理する場合は、それらのプリンタの印刷フィルタを管理する必要がある場合があります。

印刷フィルタの作成

新しい印刷フィルタを作成するには、印刷フィルタプログラムを書き、印刷フィルタの定義を作成しなければなりません。フィルタには、入力形式、出力形式、フィルタ内でコマンド行引数を処理するための言語を提供する複雑なオプションが含まれます。説明と手順については、144 ページの「新しい印刷フィルタの作成」を参照してください。

印刷フィルタの追加、変更、削除、および復元

印刷フィルタは、プリンタサーバーだけで追加、変更、または削除できます。

`lpfilter` コマンドを使用すると、利用可能なフィルタを管理できます。フィルタに関するシステム情報は、`/etc/lp/filter.table` ファイルに格納されます。`lpfilter` コマンドは、テーブルに書き出すフィルタに関する情報を、フィルタ記述

子ファイルから取得します。提供されているフィルタ記述子ファイル (PostScript のみ) は、`/etc/lp/fd` ディレクトリに入っています。実際のフィルタプログラムは、`/usr/lib/lp` ディレクトリに入っています。

LP 印刷サービスでは、定義できる印刷フィルタの数に制限はありません。使用しないフィルタを削除して LP 印刷サービスによる処理を減らすことができます。(その場合は、LP はすべてのフィルタを検査して特定の印刷要求に使用するフィルタを 1 つ見つけます。) 確信が持てない場合は、フィルタを削除しないでください。

フィルタを追加、変更、または削除すると、LP 印刷サービスによって提供されている元のフィルタの一部を上書きしたり、削除したりしてしまう可能性があります。必要に応じて元のフィルタを復元し、追加したフィルタを削除できます。

SunOS ソフトウェアには、PostScript フィルタのデフォルトセットが組み込まれています。デフォルトセットは、Solaris プリンタマネージャによってプリンタサーバーに自動的に追加されます。lpd ベースの印刷サービスで使用されていた TranScript フィルタは、SunOS にも相当するものがある場合とない場合があります。表 6-1 は、デフォルトの PostScript フィルタと、該当する TranScript フィルタが存在する場合はそのフィルタ名を示しています。

表 6-1 デフォルトの PostScript フィルタ

フィルタ (Filter)	動作	相当する TranScript
download	ダウンロードフォント	
dpost	ditroff から PostScript へ	psdit
postdaisy	daisy から PostScript へ	
postdmd	dmd から PostScript へ	
postio	PostScript プリンタへのシリアルインタフェース	pscomm
postior	プリンタとの通信	
postmd	マトリックス型グレースケールから PostScript へ	
postplot	plot から PostScript へ	psplot
postprint	simple から PostScript へ	enscript
postreverse	ページの反転または選択	psrev
posttek	TEK4014 から PostScript へ	ps4014

SunOS ソフトウェアには、次のフィルタは組み込まれていません。

- TeX
- oscat (NeWSprint™ opost)
- Enscript

Enscript の代わりに `postreverse`、`postprint`、`postio`、`dpost` の各フィルタが組み込まれています。

Solaris プリンタマネージャは、プリンタサーバーにデフォルトの PostScript フィルタを追加します。これらのフィルタでは処理できない印刷を行う必要がある場合は、カスタム印刷フィルタの作成方法については、153 ページの「新しい印刷フィルタを作成する方法」を参照してください。

▼ 印刷フィルタを追加する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. 印刷フィルタの定義に基づく印刷フィルタを追加します。

```
# lpfilter -f filter-name -F filter-def
```

<code>-f filter-name</code>	印刷フィルタ用に選択する名前
<code>-F filter-def</code>	印刷フィルタの定義名

詳細については、`lpfilter(1M)` のマニュアルページを参照してください。

印刷フィルタは、プリンタサーバーの `/etc/lp/filter.table` ファイルに追加されます。

3. 次のコマンドの出力の中の印刷フィルタについての情報をチェックして、印刷フィルタが追加されているか確認します。

```
# lpfilter -f filter-name -l
```

例 — 印刷フィルタを追加する

次の例は、`daisytroff.fd` 印刷フィルタ定義を持つ `daisytroff` 印刷フィルタを追加する方法を示しています。

```
# lpfilter -f daisytroff -F /etc/lp/fd/daisytroff.fd
```

▼ 印刷フィルタを削除する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. 印刷フィルタを削除します。

```
# lpfilter -f filter-name -x
```

`-f filter-name` 削除する印刷フィルタ名

`-x` 指定したフィルタを削除する

印刷フィルタが、プリンタサーバーの `/etc/lp/filter.table` ファイルから削除されます。

3. フィルタが削除されたことを確認します。

```
# lpfilter -f filter-name -l
```

指定した名前のフィルタがないというエラーメッセージが表示されます。

例 — 印刷フィルタを削除する

次の例は、`daisytroff` 印刷フィルタを削除する方法を示しています。

```
# lpfilter -f daisytroff -x
```

▼ 印刷フィルタに関する情報を表示する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. 印刷フィルタに関する情報を要求します。

```
# lpfilter -f filter-name -l
```

`-f filter-name` 情報を表示したい印刷フィルタ。利用できるすべての印刷フィルタに関する情報を表示するには、`filter-name` に `all` を指定する

`-l` 指定したフィルタに関する情報を表示する

指定した 1 つ以上の印刷フィルタに関する情報が表示されます。

例 — 印刷フィルタに関する情報を表示する

次の例は、`postdaisy` 印刷フィルタに関する情報を要求する方法と、それに応答して表示される情報を示しています。

```
# lpfilter -f postdaisy -l
Input types: daisy
Output types: postscript
Printer types: any
```

```
Printers: any
Filter type: slow
Command: /usr/lib/lp/postscript/postdaisy
Options: PAGES * = -o*
Options: COPIES * = -c*
Options: MODES group = -n2
Options: MODES group\=([2-9]) = -n\1
Options: MODES portrait = -pp
Options: MODES landscape = -pl
Options: MODES x\=(\.[0-9]*) = -x\1
Options: MODES y\=(\.[0-9]*) = -y\1
Options: MODES magnify\=(\.[0-9]*) = -m\1
```

次の例は、`daisytroff` フィルタに関する情報をファイルにリダイレクトする方法を示しています (そのフィルタのフィルタ定義が作成されます)。これは、うっかりフィルタ定義を削除してしまった場合に便利です。

```
# lpfilter -f daisytroff -l> daisytroff.fd
```

次の例は、システムに追加されたすべての印刷フィルタを表示する方法と、それに応答して表示される情報を示しています。

```
# lpfilter -f all -l | grep Filter
(Filter "download")
Filter type: fast
(Filter "postio")
Filter type: fast
(Filter "postior")
Filter type: fast
(Filter "postreverse")
Filter type: slow
```

フォームの管理

「フォーム」は、あらかじめ決められている形式に従って情報が印刷されている用紙です。普通紙と違って、通常、フォームにはテキストまたはグラフィックスが前もって印刷されています。フォームの一般的な例としては、企業のレターヘッド、送り状、小切手、領収書、ラベルなどがあります。

「フォーム」という用語には2つの意味があります。一つは物理的な媒体 (用紙) という意味で、もう一つは LP 印刷サービスの形式を定義するソフトウェアという意味です。

LP 印刷サービスを使用すると、フォームの使用方法を制御できます。この節では、フォームの追加、変更、削除、取り付けを行う方法と、フォームへのアクセスを制御する方法について説明します。

フォームの追加、変更、または削除

フォームを追加するときには、LP印刷サービスに指示を与えて、そのフォームを利用可能なフォームのリストに加えます。また、フォームの記述と定義に必要な情報を与えなければなりません。フォームを追加するとき、その定義を入力できますが、はじめに定義を作成しておいて、ファイルに保存しておくことをお勧めします。ファイルを編集すれば、フォーム定義を変更できます。

注 - LP印刷サービスでは、フォーム定義は提供されません。

フォームを変更するには、異なる定義を持つフォームを追加し直さなければなりません。

LP印刷サービスでは、定義できるフォームの数に制限はありません。ただし、不要なフォームは削除してください。不要なフォームがあると、印刷サービスに余計な負担をかける可能性があります。

フォームの装着

フォームを印刷するには、プリンタに給紙し、コマンドを使用してフォームを「装着」し、これによって、プリンタに送られる印刷要求がこのフォーム定義を使用して印刷されることをLP印刷サービスに通知します。複数のフォームを使用する場合など、1台のプリンタで異なる種類の印刷を行う場合には、次の作業を実行します。

- 給紙してフォームを装着する前にプリンタを使用不可にします。
- フォームの準備ができたならプリンタを再び使用可能にします。そうしないと、LP印刷サービスはプリンタでそのフォームを必要としないファイルを印刷し続けます。

フォームを装着するときには、正しく揃っているかどうかを確認してください。揃え方がフォームに対して定義されている場合は、揃え方が正しくなるようにプリンタを調整し終わるまで、フォームを装着した後でパターン印刷を繰り返すように要求できます。

プリンタに取り付けられているフォームの使用を変更または中止したい場合は、フォームを取り外してLP印刷サービスに通知しなければなりません。

フォームの追跡

LP印刷サービスにより、各プリンタにどのフォームが装着されているかを確認できます。また、フォームに印刷するときに必要な記述がなければ、LP印刷サービスが通知します。フォームの記述を作成したり、各プリンタにフォームを装着したり取り外したりするのはシステム管理者の責任です。この作業はプリンタの設定時か、LP印刷サービスからの警告時に行います。

ユーザーは印刷ジョブを印刷したいフォームを指定します。管理者は特定のフォームを装着して、フォームが使用できる状態にあり、どのプリンタに装着されているかをLP印刷サービスに伝えます。ユーザーは特定のフォームを指定することによって印刷要求を出すことができます。LP印刷サービスが要求を受け取ると、フォームの装着要求を警告メッセージとして管理者に送信します。

フォームの装着に関する警告の定義

LP印刷サービス他の警告を要求するのと同じ方法で、フォームの装着に関する警告を要求します。警告の概要については、87ページの「障害の通知の設定」を参照してください。

フォームのチェック

LP印刷サービスに対してフォームを定義し終わったら、チェックしたい情報に応じて2つのコマンドのどちらかでフォームの定義をチェックできます。

- `lpforms` コマンドを使用してフォームの属性を表示します。また、コマンドの出力をファイルにリダイレクトして将来の参照に備えて保存できます。
- `lpstat` コマンドを使用してフォームの現在の状態を表示します。内容を保護するため、位置揃えパターンは表示されません。

既存のフォーム名がわからない場合は、`/etc/lp/forms` ディレクトリの内容の一覧を表示して調べることができます。

フォームへのアクセスの制限

どのプリンタやユーザーが、ネットワーク上で利用可能な一部またはすべてのフォームを使用できるかを制御できます。たとえば、経理部に属するユーザーだけが小切手のフォームを印刷できるようにしたい場合があります。また、特定のプリンタだけで利用できる小切手のフォームが必要な場合もあります。

フォームへのユーザーアクセスを制限するには、126ページの「フォームへのユーザーアクセスを制限する方法」、フォームへのプリンタアクセスを制限するには、127ページの「フォームへのプリンタアクセスを制限する方法」を参照してください。

▼ フォームを追加する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. フォーム定義に基づくフォームを追加します。

フォームが `/etc/lp/forms/form-name` ファイルから削除されます。

3. フォームが削除されたことを確認します。

```
# lpforms -f form-name -l
```

指定したフォーム名がないことを示すエラーメッセージが表示されます。

例 — フォームを削除する

次の例は、`medical` フォームを削除する方法を示しています。

```
# lpforms -f medical -x
```

▼ フォームを取り外し、装着する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. 現在のフォームを取り外そうとしているプリンタで印刷要求の受け入れを停止します。

```
# reject printer-name
```

printer-name フォームを取り外すプリンタ名

新しい印刷要求 (フォームを必要としない場合もある) は、そのプリンタの待ち行列に入れられなくなります。

3. 現在のフォームを取り外します。

```
# lpadmin -p printer-name -M -f none
```

このコマンドの変数 `printer-name` は、手順 2 と同じです。

現在のフォームは、プリンタサーバーの `/etc/lp/printers/printer-name/configuration` ファイルから削除されます。

4. プリンタからフォーム用紙を取り外します。
5. 次の印刷要求のためにフォーム用紙を装着します。
6. フォームを装着します。

```
# lpadmin -p printer-name -M -f form-name [-a -o filebreak]
```

`-p printer-name` フォームを装着するプリンタ名

<code>-M -f form-name</code>	装着するフォーム名
<code>-a -o filebreak</code>	フォームに位置揃えパターンが定義されている場合は、そのコピーを印刷できるようにする

指定したフォームは、プリンタサーバーの `/etc/lp/printers/printer-name/configuration` ファイルに追加されます。

7. プリンタ上で印刷要求の受け付けを開始します。

```
# accept printer-name
これで、プリンタは新しく装着したフォームで印刷する準備ができました。
```

8. 次のコマンド出力の「Form mounted」見出しの下にあるフォーム名をチェックし、フォームが装着されていることを確認します。

```
$ lpstat -p printer-name -l
あるいは、新しいフォームを必要とする印刷要求を出して、プリンタの出力をチェックします。
```

例 — フォームを取り外し、装着する

次の例は、現在装着されているフォームをプリンタ luna から取り外すプロセスを示しています。

```
# reject luna
destination "luna" will no longer accept requests
# lpadmin -p luna -M f none
# accept luna
destination "luna" now accepting requests
```

次の例は、medical フォームをプリンタ luna 上に装着するプロセスを示しています。

```
# reject luna
destination "luna" will no longer accept requests
# lpadmin -p luna -M f medical -a -o filebreak
# accept luna
destination "luna" now accepting requests
```

▼ フォームの装着に関する警告を設定する方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. フォームの装着に関する要求警告を設定します。

```
# lpforms -f form-name -A alert [-Q requests] [-W minutes]
```

<code>-f form-name</code>	要求警告を設定したいフォーム名
<code>-A alert</code>	フォームが要求されるときに出す警告の種類を指定する。 <code>alert</code> に有効な値については、表 5-3 を参照。有効な値は <code>mail</code> 、 <code>write</code> 、 <code>quiet</code> など。 <code>mail</code> または <code>write</code> を選択すると、あらかじめ定義された警告メッセージが表示される。この警告メッセージは、指定されたフォームの装着を促すもので、そのフォームを使用するように設定されている 1 つ以上の複数のプリンタ名が含まれる
<code>-Q requests</code>	警告が出される前に、フォームが必要な印刷要求がいくつ待ち行列に入っていないかを指定する。このオプションを指定しなければ、印刷要求が待ち行列に 1 つ入っただけで警告が出される
<code>-W minutes</code>	警告が出される間隔 (分単位) を指定する。このオプションを指定しなければ、警告は一度だけ送信される

要求警告は、プリンタサーバーの `/etc/lp/forms/form-name/alert.sh` ファイルに追加されます。

- 次のコマンドの出力をチェックして、そのフォームに関する警告が追加されていることを確認します。

```
# lpforms -f form-name -A list
```

あるいは、警告を出すために印刷要求の低い番号を設定した場合、最小限の要求を満たすために十分な印刷要求を出し、フォームの装着を促す警告を受け取ることを確認します。

例 — フォームの装着に関する警告を設定する

次の例は、印刷待ち行列に `letterhead` フォームに関する 10 の印刷要求があるときに、`letterhead` に関して 5 分ごとに電子メールで警告が送られるように設定する方法を示します。

```
# lpforms -f letterhead -A mail -Q 10 -W 5
```

次の例は、印刷待ち行列に `letterhead` フォームに関する 5 の印刷要求があるときに、`letterhead` に関して 10 分ごとにコンソールウィンドウに警告が送られるように設定する方法を示します。

```
# lpforms -f letterhead -A write -Q 5 -W 10
```

次の例は、`invoice` フォームに関して要求警告が送られないように設定する方法を示しています。

```
# lpforms -f invoice -A none
```

▼ フォームに関する情報を表示する方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. フォームに関する情報を要求します。

```
# lpforms -f form-name -l
```

`-f form-name` 情報を表示したいフォーム名。利用できるすべてのフォームに関する情報を表示するには、`form-name` に `all` を指定する

`-l` 指定したフォームを表示する

指定した 1 つ以上のフォームに関する情報が表示されます。

例 — フォームに関する情報を表示する

次の例は、`medical` フォームに関する情報を表示する方法を示しています。

```
# lpforms -f medical -l
Page length: 62
Page width: 72
Number of pages: 2
Line pitch: 6
Character pitch: 12
Character set choice: pica
Ribbon color: black
Comment:
Medical claim form
```

次の例は、`medical` フォームに関する情報をファイルにリダイレクトするコマンドを示しています (このコマンドは、そのフォームのフォーム定義を作成します)。これは、うっかりフォーム定義を削除してしまった場合に便利です。

```
# lpforms -f medical -l> medical.fmd
```

▼ フォームの現在の状態を表示する方法

1. プリンタサーバーにログインします。
2. フォームの現在の状態に関する情報を要求します。

```
$ lpstat -f form-name
```

`-f form-name` 現在の状態を表示したいフォーム名。すべてのフォームの現在の状態を表示したい場合は、`form-name` に `all` を指定する

指定した 1 つ以上のフォームの現在の状態に関する情報が表示されます。

例 — フォームの現在の状態を表示する

次の例は、`medical` フォームの状態を表示する方法を示しています。

```
$ lpstat -f medical
form medical is available to you
```

▼ フォームへのユーザーアクセスを制限する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. フォームへのユーザーアクセスを許可または拒否します。

```
# lpforms -f form-name -u allow:user-list | deny:user-list
```

<code>-f form-name</code>	ユーザーアクセスの許可または拒否リストを作成するためのフォーム名
<code>-u allow: user-list</code>	ユーザーアクセスの許可リストに追加するユーザー名。複数のユーザーログイン名を指定する場合は、空白またはコンマで区切る。空白で区切る場合は、ID のリストを引用符で囲む。 <i>user-list</i> の有効な値については、表 5-5 を参照
<code>deny: user-list</code>	ユーザーアクセス拒否リストに追加するユーザー名。複数のユーザーログイン名を指定する場合は、空白またはコンマで区切る。空白で区切る場合は、ID のリストを引用符で囲む。 <i>user-list</i> の有効な値については、表 5-5 を参照

プリンタサーバーの次のどちらかのファイルの指定されたフォーム用の許可または拒否のユーザーアクセスリストに、指定した 1 人以上のユーザーが追加されます。

```
/etc/lp/forms/form-name/allow または
/etc/lp/forms/form-name /deny
```

3. ユーザーアクセスの許可および拒否リストを確認します。

```
# lpforms -f form-name -l
```

例 — フォームへのユーザーアクセスを制限する

次の例は、ユーザー `nathan` と `marcia` にのみ `check` フォームへのアクセスを許可する方法を示しています。

```
# lpforms -f check -u allow:nathan,marcia
```

次の例は、ユーザー `jones` と `smith` による `dental` フォームへのアクセスを拒否する方法を示しています。

```
# lpforms -f dental -u deny:"jones,smith"
```

▼ フォームへのプリンタアクセスを制限する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. プリンタでのフォームの使用を許可または拒否します。

```
# lpadmin -p printer-name -f allow:form-list | deny:form-list
```

<code>-p printer-name</code>	フォームの許可リストまたは拒否リストを作成するプリンタ名
<code>-f allow: form-list deny:form-list</code>	許可または拒否リストに追加されるフォーム名。複数のフォーム名は空白またはコンマで区切る。空白で区切る場合は、フォーム名のリストを引用符で囲む

指定した1つ以上のフォームは、プリンタサーバーの次のどちらかのファイルの許可または拒否フォームリストに追加されます。

```
/etc/lp/printers/printer-name /form.allow
```

```
/etc/lp/printers/printer-name /form.deny
```

3. 許可および拒否フォームリストを確認します。

```
# lpstat -p printer-name -l
```

例 — フォームへのプリンタアクセスを制限する

次の例は、プリンタ `luna` に `medical`、`dental`、`check` の各フォームへのアクセスのみを許可する方法を示しています。

```
# lpadmin -p luna -f allow:medical,dental,check
```

次の例は、プリンタ `luna` による `medical`、`dental`、`check` の各フォームへのアクセスを拒否する方法を示しています。

```
# lpadmin -p luna -f deny:"medical dental payroll"
```

フォントの管理

レーザープリンタがある場合は、PostScript 用のフォントをインストールして管理する必要がある可能性があります。PostScript フォントをインストールするシステムと、その管理方法を決定する必要がある可能性があります。多くのプリンタの場合、プリンタのインストール作業の一部としてフォントを設定します。

PostScript フォントは、プリンタかプリンタと通信を行うシステムのどちらかに、アウトライン形式で格納されます。文書の印刷時に、PostScript インタプリタは、アウトライン記述から適切な大きさの各文字を必要に応じて生成します。文書に必要なフォントが使用するプリンタに格納されていない場合は、文書が印刷される前にそのフォントをプリンタに転送しなければなりません。この転送処理を「フォントのダウンロード」といいます。

フォントは、次のいくつかの方法で格納または使用されます。

- 「プリンタ常駐フォント」は、プリンタに常時格納されています。これらのフォントは、メーカーによってプリンタの読み取り専用メモリー (ROM) にインストールされている場合があります。プリンタがディスクを備えている場合は、そのディスクにフォントをインストールしなければならないことがあります。ほとんどの PostScript プリンタには、35 の標準フォントが付いています。
- 「常時ダウンロードフォント」は、PostScript の exitserver プログラムを使用してプリンタに転送されます。常時ダウンロードフォントは、プリンタの電源を切るまでプリンタのメモリーに残っています。ダウンロードフォントに割り当てられたメモリーによって、PostScript 印刷要求では利用可能なサーバーのメモリーが減少します。exitserver プログラムを使用するには、プリンタシステムのパスワードが必要で、プリンタ管理者が使用します。プリンタで出力する大部分の印刷要求に特定のフォントを使用するときは、そのフォントを常時ダウンロードするようにしてください。
- あまり頻繁に使用されないか、特殊な目的で使用されるフォントは、ユーザーのシステムに格納できます。ユーザーは、印刷要求を出すときにこれらのフォントを指定できます。そのフォントは印刷要求に追加されてプリンタに転送されます。印刷要求が処理されると、フォントに割り当てられたメモリー空間は、他の印刷要求が使用できるように解放されます。
- 「ホスト常駐フォント」は、多数のユーザーによって共有されるシステムに格納されます。フォントを格納するシステムは、プリンタサーバーでも印刷クライアントでもかまいません。各ユーザーは印刷する文書のフォントを要求指定できます。この方法は、多数のフォントを利用できるときや、これらのフォントがすべての印刷要求で使用されるとは限らないときに便利です。そのフォントがプリンタサーバーに接続されたプリンタでのみ使用される場合は、プリンタサーバーに格納してください。そのフォントが1つのシステム上で、ネットワーク上の複数のプリンタに要求を依頼する可能性があるユーザーによって使用される場合は、フォントはそのユーザーのシステムに格納してください。

LP 印刷サービスには、ホスト常駐フォントを管理するための特殊なダウンロード用のフィルタがあります。また、troff プログラムで使用するために、多くの PostScript プリンタに搭載された 35 の標準 PostScript フォント用の troff (1) のフォント幅テーブルも提供しています。

プリンタ常駐フォントの管理

ほとんどの PostScript プリンタは、プリンタ内蔵の ROM にフォントが搭載されています。プリンタによっては、追加フォントを格納するためのディスクが用意されています。プリンタをインストールするときに、そのプリンタ用のフォントリストにプリンタ常駐フォントを追加してください。プリンタ常駐フォントがわかっている場合、フォントをネットワーク経由で必要以上に転送することがなくなります。各プリンタには搭載されているフォントの独自のリストがあります。

```
/etc/lp/printers/printer-name/residentfonts
```

プリンタをプリンタサーバーに接続するときには、プリンタサーバー上においてプリンタにダウンロードできるフォントが、residentfonts ファイル内のリストに含まれているかどうかを確認します。

プリンタ常駐フォントのリストが入っているファイルは、vi などのテキストエディタを使用して編集しなければなりません。

ホスト常駐フォントのダウンロード

PostScript の文書がプリンタにダウンロードされていないフォント指定を含んでいるときは、「ダウンロードフィルタ」がこの印刷要求を管理します。ダウンロードフィルタは PostScript の文書作成規則を使用して、ダウンロードするフォントを決定します。

LP 印刷フィルタには、高速フィルタと低速フィルタがあります。高速フィルタは、印刷するファイルをすばやく準備し、フィルタが処理している間にプリンタにアクセスしなければなりません。低速フィルタはファイルの変換に時間がかかり、フィルタが処理している間にプリンタにアクセスする必要はありません。低速フィルタの例には、ASCII ファイルから PostScript ファイルへのフィルタがあります。

ダウンロードフィルタは高速フィルタです。フォントがプリンタサーバー上にある場合は、フォントを自動的にダウンロードします。また、ダウンロードフィルタを使用して、プリンタサーバーにフォントを転送することもできます。そのためには、lp -y コマンドを指定して、ダウンロードフィルタを低速フィルタとして呼び出すための新しいフィルタテーブルのエントリを作成します。あるいは、入力タイプを変更して、このフィルタの選択を強制することもできます。

ダウンロードフィルタは、次の 5 つの作業を実行します。

1. PostScript の文書を検索して、要求されているフォントを判別します。これらの要求は、ヘッダコメントの PostScript 構造化コメント %%DocumentFonts:font1 font2 ... で指定されます。

2. プリンタ常駐フォントのリストを検索して、要求されたフォントをダウンロードしなければならないかどうかを判別します。
3. フォントがプリンタ上になれば、ダウンロードフィルタは(マップテーブルから適切なファイル名を読み取って) ホスト常駐フォントのディレクトリを検索し、要求されたフォントが利用可能かどうかを判別します。
4. そのフォントが利用可能であれば、フィルタはそのフォントのファイルを取り出し、印刷するファイルに追加します。
5. フォント定義ファイルとソースファイル(印刷するファイル)を PostScript プリンタに送ります。

ホスト常駐フォントのインストールと管理

フォントによっては、ホストシステムに格納されており、特定の印刷要求に応じてプリンタに転送されるものがあります。管理者は、システム上のすべてのユーザーが PostScript フォントを使用できるように管理する必要があります。そのためには、これらのフォントのインストール方法とインストール場所を知っておかなければなりません。フォントは名前で要求され、ファイルに格納されているので、LP 印刷サービスはフォント名とフォントを定義しているファイル名を対応付けるマップファイルを持っています。ホスト常駐フォントをインストールするときには、マップファイルとフォントリストの両方を更新しなければなりません。

PostScript プリンタで利用できるフォントは、ユーザーが作成した `/usr/share/lib/hostfontdir/typeface/font` ディレクトリに格納されます。この場合、`typeface` は `palatino` や `helvetica` などの名前に置き換えられ、`font` は `bold` や `italic` などの名前に置き換えられます。

▼ ダウンロードされた PostScript フォントをインストールする方法

1. プリンタサーバーまたは印刷クライアントにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. `/etc/lp/printers/printer-name` ディレクトリに変更します。

```
# cd /etc/lp/printers/printer-name
```

`printer-name` ダウンロードされた PostScript フォントをインストールするプリンタ名

3. `residentfonts` ファイルが存在しない場合は作成します。

```
# touch residentfonts
```

常駐させるダウンロードフォントを初めて追加する場合は、このファイルが存在しないことがあります。

4. `residentfonts` ファイルを編集して、すべてのプリンタ常駐フォントとダウンロードフォントを追加します。

▼ ホスト常駐 PostScript フォントをインストールする方法

1. プリンタサーバーまたは印刷クライアントにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。

2. `hostfontdir` ディレクトリが存在しない場合は作成します。

```
# cd /usr/share/lib
# mkdir hostfontdir
# chmod 775 hostfontdir
```

3. 新しい書体のディレクトリが存在しない場合は作成します。

```
# mkdir typeface
```

4. フォントファイルを適切なディレクトリにコピーします。

```
# cp filename /usr/share/lib/hostfontdir/typeface/font
```

5. マップテーブルに組み込むフォント名とファイル名を追加します。

- a. `/usr/share/lib/hostfontdir` ディレクトリに変更します。

- b. `vi` などのテキストエディタを使用して `map` ファイルを編集します。

テーブルに追加したいフォントごとに1行ずつエントリを追加します。エントリには、フォント名、スペース1個、フォントが常駐するファイル名の順に入力します。たとえば、次のようになります。

```
Palatino-Bold /usr/share/lib/hostfontdir/palatino/bold
```

- c. ファイルを保存します。

適切なシステム上のマップテーブルにサンプルエントリを入れておけば、ユーザーは各自の印刷ジョブに(たとえば、`Palatino Bold` などの)フォントを適用できます。このフォントを含む印刷要求を依頼すると、LP印刷サービスはそのファイルに `/usr/share/lib/hostfontdir/palatino/bold` のコピーを追加してから、プリンタに送信します。

6. `troff` を使用している場合は、このフォント用の新しいフォント幅テーブルを標準 `troff` フォントディレクトリ内で作成します。

第 7 章

LP 印刷サービスのカスタマイズ (手順)

この章では、LP 印刷サービスのカスタマイズについての概要と手順について説明します。

この章で説明する手順は次のとおりです。

- 134 ページの「プリンタポート特性を調整する方法」
- 138 ページの「サポートされていないプリンタの terminfo エントリを追加する方法」
- 142 ページの「独自のプリンタインタフェースプログラムを設定する方法」
- 153 ページの「新しい印刷フィルタを作成する方法」
- 157 ページの「新しいフォーム定義を作成する方法」

プリンタの概要については、第 2 章を参照してください。

プリンタポート特性の調整

LP 印刷サービスによって設定されるプリンタポート特性には、プリンタの通信設定と互換性がなければなりません。LP 印刷サービスから提供されたデフォルトのプリンタポート設定値がプリンタで機能しない場合は、プリンタのマニュアルを参照し、そのプリンタが LP 印刷サービスに対してどのような設定値が必要かを調べてください。プリンタ通信設定を設定および表示するには、`stty` コマンドを使用します。

表 7-1 は、LP 印刷サービスに使用されるデフォルトの `stty` 設定値を示しています。

表 7-1 LP 印刷サービスに使用される `stty` のデフォルト設定値

オプション	説明
-9600	ボーレートを 9600 に設定する


```
-o "stty=options"
```

options で指定したポート特性 (stty オプション) を設定する。このコマンドで1つ以上の stty オプションを変更できる。オプションを区切るには空白を使用し、オプションを多重引用符で囲む。オプションの全リストについては、stty (1) を参照。LP 印刷サービスで使用されるデフォルトの stty 設定については、表 7-1 を参照

3. プリンタポート特性が変更されたことを確認します。

```
# stty -a
```

例 — プリンタポート特性を調整する

次の例は、プリンタ luna のポート特性を設定する方法を示しています。parenb オプションはパリティチェック/生成を使用可能にし、parodd は奇数パリティの生成を設定し、cs7 は文字サイズを7ビットに設定します。

```
# lpadmin -p luna -o "stty='parenb parodd cs7'"
```

次の例は、プリンタ venus の端末ボーレートを 19200 に設定する方法を示しています。

```
# lpadmin -p venus -o "stty=19200"
```

サポートされていないプリンタの terminfo エントリを追加する

LP 印刷サービスは、インタフェースプログラムと terminfo データベースを使用して各プリンタを初期化し、選択されたページサイズ、文字ピッチ、行ピッチ、および文字セットを設定します。

各プリンタは、terminfo データベース内で短縮名を使用して識別されます。terminfo データベースで使用される名前は、TERM シェル変数を設定するのに使用する名前と同じです。また、この名前はプリンタを設定するときに指定するプリンタタイプでもあります。たとえば、各種の PostScript プリンタのエントリは、`/usr/share/lib/terminfo/p` ディレクトリに入っています。SunOS システムに組み込まれているデフォルトエントリは、PS (PostScript 用) と PSR (Reverse PostScript 用) です。

プリンタの terminfo エントリが見つからなくても、ページサイズ、ピッチ、および文字セットを自動選択しないで、LP 印刷サービスでプリンタを使用できます。ただし、印刷要求ごとに正しいモードでプリンタが設定されないという問題が起きることがあります。

使用中のプリンタタイプの terminfo エントリがないが、プリンタを正しいモードに設定しておきたい場合は、プリンタで使用するインタフェースプログラムをカスタマイズするか、terminfo データベースにエントリを追加できます。terminfo データベース内の端末エントリやプリンタエントリには、多数の項目が入っており、定義されています。しかし、LP 印刷サービスはそのうち 50 未満の項目しか使用しません。表 7-2 は、プリンタに必要な terminfo 項目を示しています。

表 7-2 プリンタに必要な terminfo 項目

項目	説明
ブール値	
cpix	文字ピッチを変更すると解像度が変わる
daisy	プリンタで文字セットを変更するには演算子が必要
lpix	行ピッチを変更すると解像度が変わる
数値	
bufsx	印刷前にバッファされるバイト数
cols	1 行の列数
cps	1 秒あたりの文字の平均印刷速度
it	最初はスペース n 個分ごとのタブ
lines	1 ページの行数
orc	1 文字あたりの水平解像度
orhi	1 インチあたりの水平解像度
orl	1 行あたりの垂直解像度
orvi	1 インチあたりの垂直解像度
文字列	
chr	水平解像度を変更する
cpi	1 インチあたりの文字数を変更する
cr	キャリッジリターン
csnm	文字セット名のリスト
cudl	1 行下げる
cud	キャリッジを n 行下に移動する
cuf	キャリッジを n 列右に移動する
cvr	垂直解像度を変更する

表 7-2 プリンタに必要な terminfo 項目 (続き)

項目	説明
ff	紙送りする
hpa	水平絶対位置
ht	次の 8 スペースタブストップまでのタブ
if	初期化ファイル名
iprogr	初期化プログラムのパス名
is1	プリンタ初期化文字列
is2	プリンタ初期化文字列
is3	プリンタ初期化文字列
文字列	
lpi	1 インチあたりの行数を変更する
mgc	マージン (上、下、横) をすべて消去する
rep	文字を n 回繰り返す
rwidm	倍幅の印刷を使用不可にする
scs	文字セットを選択する
scsd	文字セットの定義を開始する
slines	ページの長さを 1 ページあたり n 行に設定する
smgl	現在の列の左マージンを設定する
smglp	左マージンを設定する
smgr	現在の列の右マージンを設定する
smgrp	右マージンを設定する
smglr	左右のマージンを設定する
msgt	現在の行の上マージンを設定する
smgtp	上マージンを設定する
smgb	現在の行の下マージンを設定する
smgbp	下マージンを設定する
smgtb	上下のマージンを設定する
swidm	倍幅の印刷を使用可能にする
vpa	垂直絶対位置

▼ サポートされていないプリンタの terminfo エントリを追加する方法

注 - プリンタの terminfo エントリを作成する前に、まず、そのプリンタをサポートする既存の terminfo エントリがないことを確認してください。そのためには、類似するプリンタがあれば、そのエントリを使用してプリンタを設定してください。

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。

2. プリンタの terminfo エントリ名を決定します。

/usr/share/lib/terminfo ディレクトリ内のディレクトリには、有効な terminfo エントリがすべて入っています。それを参考にしてプリンタ名を選択してください。

3. プリンタの terminfo エントリファイルを作成します。

LP 印刷サービスに新しいプリンタを追加するために terminfo エントリ内で定義しなければならない項目については、表 7-2 を参照してください。terminfo データベースの構造については、terminfo(4) のマニュアルページを参照してください。新しい terminfo エントリを作成しやすいように、infocmp コマンドを使用して既存の terminfo エントリをファイルに保存します。これは、作成したいエントリに似た terminfo エントリがある場合に便利な方法です。たとえば、次のコマンドで ps エントリを ps_cust ファイルに保存すると、新しい terminfo エントリになります。

```
infocmp ps > ps_cust
```

4. terminfo エントリをコンパイルして terminfo データベースに入れます。

```
# tic terminfo_entry
```

```
terminfo_entry          作成した terminfo エントリファイル
```

5. /usr/share/lib/terminfo ディレクトリ内で新しい terminfo エントリファイルをチェックします。

プリンタインタフェースプログラムのカスタマイズ

標準プリンタインタフェースプログラムでサポートされないプリンタを使用する場合は、独自のプリンタインタフェースプログラムを提供できます。標準プログラムをコピーし、指定したプリンタを使用するように LP 印刷サービスに指示できます。ただし、まず標準プログラムについて理解する必要があります。次の節では、標準プログラムについて説明します。

プリンタインタフェースプログラムの機能は次のとおりです。

- 必要に応じてプリンタポートを初期化する。標準プリンタインタフェースプログラムは、`stty` コマンドを使用してプリンタポートを初期化する。
- プリンタハードウェアを初期化する。標準プリンタインタフェースプログラムは、`terminfo` データベースと `TERM` シェル変数から制御シーケンスを取得する。
- 必要に応じてバナーページを印刷する。
- 印刷要求で指定された部数を印刷する。



注意 – リリース 3.2 より前の UNIX System V のプリンタインタフェースプログラムを使用している場合でも、そのプログラムは、SunOS 5.9 または互換 LP 印刷サービスで使用できます。ただし、一部の `-o` オプションは SunOS 5.9 または互換 LP 印刷サービスでは標準化されていて、すべてのプリンタインタフェースプログラムに渡されます。これらのオプションは、古いインタフェースで使用される同じ名前のオプションの妨げとなることがあります。

プリンタポートを開く処理は、プリンタインタフェースプログラムではなく LP 印刷サービスが受け持ちます。プリンタポートは標準出力としてプリンタインタフェースプログラムに与えられ、プリンタはプリンタインタフェースプログラムの「制御端末」として識別されるので、ポートが「ハング」するとプリンタインタフェースプログラムに `SIGHUP` 信号が送信されます。

標準プリンタインタフェースプログラム

LP 印刷サービスは、標準 (モデル) プリンタインタフェースプログラム `/usr/lib/lp/model/standard` を使用して、表 7-3 の印刷デフォルトを設定します。

表 7-3 デフォルトのプリンタポート特性

特性	デフォルト設定
デフォルトフィルタ	None
文字ピッチ	None
行ピッチ	None
ページ幅	None
ページ長	None
文字セット	None
stty オプション	9600 cs8 -cstopb -parenb -parodd ixon -ixany opost -olcuc onlcr -ocrnl -onocr -onlret -ofill nl0 cr0 tab0 bs0 vt0 ff0
終了コード	0

stty モードのカスタマイズ

ボーレートや出力オプションなどの端末特性を変更する必要がある場合は、標準プリンタインタフェースプログラム内で、次のコメントから始まるセクションを探します。

```
## Initialize the printer port
```

終了コード

印刷し終わると、インタフェースプログラムは印刷ジョブの状態を示すコードを返して終了します。終了コードは、プリンタインタフェースプログラムの最後のエントリです。

表 7-4 は、終了コードとそのコードが LP 印刷サービスでどのように解釈されるかを示しています。

表 7-4 プリンタインタフェースプログラムの終了コード

コード	LP 印刷サービスにとっての意味
0	印刷要求は正常に完了した。プリンタ障害が発生した場合は、クリアされた

表 7-4 プリンタインタフェースプログラムの終了コード (続き)

コード	LP 印刷サービスにとっての意味
1 から 127	要求の印刷中に問題が発生した (たとえば、印字できない文字が多すぎる、要求がプリンタの容量を超えているなど)。LP 印刷サービスは、その要求を依頼したユーザーに、印刷中にエラーが発生したことを通知する。このエラーはその後の印刷要求には影響しない。プリンタ障害が発生するとクリアされる
128	このコードは、LP 印刷サービスが内部で使用するために予約されている。インタフェースプログラムは、このコードを返して終了してはいけない
129	要求の印刷中にプリンタ障害が発生した。この障害は、その後の印刷要求に影響を及ぼす。プリンタの障害回復が LP 印刷サービスに管理者によって問題が解決されるまで待つように指示すると、LP 印刷サービスはプリンタを使用不可にする。障害回復後に印刷を続けようとする、LP 印刷サービスはプリンタを使用不可にしないが、数分後にそのまま印刷しようとする
>129	これらのコードは、LP 印刷サービスが内部で使用するために予約されている。インタフェースプログラムは、この範囲内のコードを返して終了してはいけない

プログラムがコード 129 を返して終了すると、root はプリンタ障害を警告されます。また、LP 印刷サービスは、障害がクリアされた後に要求を最初から印刷し直さなければなりません。要求全体を印刷し直したくない場合は、インタフェースプログラムに障害メッセージを LP 印刷サービスへ送信させることもできますが、障害がクリアされるまで待つこととなります。障害がクリアされると、インタフェースプログラムはファイルの印刷を再開できます。印刷が終了すると、プリンタインタフェースプログラムは障害が発生しなかった場合と同様に終了コード 0 を返すことができます。このアプローチには、障害が自動的にクリアされた場合に、それをインタフェースプログラムが検出できるので、管理者がプリンタを再び使用可能にする必要がないという利点もあります。

障害メッセージ

lp.te11 プログラムを使用すると、LP 印刷サービスに障害メッセージを送信できます。このプログラムは、標準プリンタインタフェースコード内の LPTELL シェル変数によって参照されます。プログラムは標準入力を取り込んで LP 印刷サービスに送信し、LP 印刷サービスは管理者にプリンタ障害を警告するメッセージを出します。標準入力が空であれば、lp.te11 は警告を開始しません。lp.te11 プログラムの例として、次のコメントの直後の標準プリンタインタフェースコードを確認してください。

```
# Set up the $LPTELL program to capture fault messages here
```

特殊な終了コード 129 または `lp.tell` プログラムを使用すると、プリンタインタフェースプログラムはプリンタ自体を使用不可にする必要がありません。インタフェースプログラムは、プリンタを直接使用不可にできますが、その場合は障害警告メカニズムが無効になります。LP 印刷サービスがプリンタ障害を検出した場合にのみ警告が送信され、特殊終了コードと `lp.tell` プログラムはその主要検出ツールです。

LP 印刷サービスがいずれかの時点でファイルの印刷を中断しなければならない場合は、TERM 信号 (トラップ番号 15) を使用してインタフェースプログラムを終了させます。(kill(1) および signal(3C) のマニュアルページを参照)。プリンタインタフェースプログラムが他の信号を受信して終了した場合は、LP 印刷サービスはその後の印刷要求は影響されないものとみなし、そのプリンタを使用し続けます。LP 印刷サービスは、要求を依頼したユーザーに、その要求が正常に終了しなかったことを通知します。

インタフェースが最初に呼び出されると、信号 HUP、INT、QUIT、PIPE (トラップ番号 1、2、3、13) は無視されます。標準インタフェースは、信号が適切な時期にトラップされるように、この動作を変更します。標準インタフェースはこれらの信号の受信をプリンタの問題を示す警告として解釈し、信号を受信すると障害警告を発生します。

カスタマイズされたプリンタインタフェースプログラムの使用方法

カスタマイズされたプリンタインタフェースプログラムを作成し、プリンタサーバー上で標準プリンタインタフェースプログラムの代わりに使用できます。そのためには、`lpadmin` コマンドを使用して、プログラムを特定のプリンタの LP 印刷サービスに登録します。

▼ 独自のプリンタインタフェースプログラムを設定する方法

1. プリンタサーバーにスーパーユーザーまたは `lp` としてログインするか、同等の役割になります。
2. 独自のプリンタインタフェースプログラムがあるかどうかに基づいて次の手順を選択します。

カスタムプリンタインタフェースの状態	次の手順
独自のプリンタインタフェースプログラムがないので作成する必要がある	手順 3 へ

カスタムプリンタインタフェースの状態	次の手順
すでに独自のプリンタインタフェースプログラムがある	手順 5へ

- 標準プリンタインタフェースプログラムをコピーします。

```
# cp /var/spool/lp/model/standard custom-interface
```

- 標準プリンタインタフェースプログラムのコピーを必要に応じて変更します。
詳しくは、139 ページの「標準プリンタインタフェースプログラム」のプログラムの説明を参照して、何を変更する必要があるかを判断してください。
- 特定のプリンタ独自のプリンタインタフェースプログラムを設定します。

```
# lpadmin -p printer-name -i custom-interface
```

<code>-p printer-name</code>	独自のプリンタインタフェースプログラムを使用するプリンタ名
<code>-i custom-interface</code>	独自のプリンタインタフェースプログラム名

独自のプリンタインタフェースプログラムが LP 印刷サービスに登録され、ユーザーが印刷要求を出すと、そのプリンタに使用されます。

- 独自のプリンタインタフェースプログラムが `/etc/lp/printers/printer-name/configuration` ファイルに追加されたかどうかを確認します。

例 — 独自のプリンタインタフェースプログラムを設定する

次の例は、プリンタ `luna` の独自のプリンタインタフェースプログラム `custom` を設定する方法を示しています。

```
# lpadmin -p luna -i custom
```

次の例は、システム `venus` がプリンタ `asteroid` 上で使用している独自のプリンタインタフェースプログラムを設定する方法を示しています。

```
# lpadmin -p asteroid -e venus
```

新しい印刷フィルタの作成

フィルタは、LP 印刷サービスがプリンタで解釈できないタイプのファイルを印刷する必要があるときに使用されます。新しい印刷フィルタを作成するのは簡単ではありません。通常は広範囲の経験が必要です。新しい印刷フィルタを定義するには、次の 2 つの手順があります。

- 印刷フィルタプログラムを作成する
- 印刷フィルタ定義を作成する

印刷フィルタは、必要に応じて簡単なものでも複雑なものでもかまいません。フィルタには、入力形式、出力形式、そのフィルタ内でコマンド行引数を処理する言語を提供する複雑なオプションが入っています。

PostScript 以外のプリンタを使用する場合は、必要に応じて印刷フィルタを作成して追加する必要があります。まず、印刷フィルタの機能と、フィルタプログラムが満たさなければならない要件を理解しておく必要があります。

印刷フィルタプログラムの作成

LP 印刷サービスには、`/usr/lib/lp/postscript` ディレクトリにフィルタプログラムが組み込まれています。これらのフィルタは、宛先プリンタが PostScript 形式のデータを必要とするほとんどの PostScript 印刷の状況に対応します。印刷フィルタプログラムは、2 進の実行可能プログラムでなければなりません。

フィルタのタイプ

印刷フィルタには、高速フィルタと低速フィルタの 2 種類があります。

高速フィルタは、ファイルの印刷準備にあまり処理時間がかかりません。また、実行するときにはプリンタにアクセスしなければなりません。印刷障害を検出する印刷フィルタは、高速フィルタでなければなりません。PRINTER キーワードをフィルタオプションとして使用するフィルタは、高速フィルタとしてインストールしなければなりません。

低速フィルタは、ファイルの印刷準備に大量の処理時間がかかります。また、低速フィルタは実行するときにはプリンタにアクセスする必要がありません。低速フィルタはバックグラウンドで実行されるので、プリンタと連結する必要がなく、低速フィルタを必要としない他のファイルを印刷できます。

ファイルの変換

LP 印刷サービスは、印刷フィルタを使用して、ある内容形式から別の内容形式にファイルを変換します。プリンタごとに、受け付けられるファイル内容形式を指定できます。ユーザーは印刷要求を出すときにファイル内容形式を指定し、LP 印刷サービスはその内容形式のファイルを印刷できるプリンタを探します。多くのアプリケーションは各種プリンタに合わせてファイルを生成できるので、通常はこれで十分です。ただし、アプリケーションによっては、利用できるプリンタ上で印刷できないファイルを生成するものがあります。

LP 印刷サービスは、プリンタで直接受け付けられない形式のファイルの印刷要求を受信するたびに、印刷要求の内容形式を利用できる (または指定された) プリンタの内容形式と一致させようとします。一致すると、そのファイルはフィルタをかけなくてもプリンタに直接送信できます。一致する形式が見つからない場合や、内容形式でフィルタを使用するように指定されている場合は、LP 印刷サービスはファイルの内容形式を利用できるフィルタの入力内容形式と一致させ、フィルタの出力形式をプリンタの内容形式と一致させようとします。適切なフィルタが見つかると、印刷要求はフィルタを通して渡されます。

特殊印刷モードの処理

印刷フィルタは、特殊モードを処理し特殊ページの印刷を要求します。特殊印刷モードは、カスタマイズされたフィルタが必要な印刷要求の特性を印刷するために必要です。フィルタは、次の特性を処理します。

- プリンタタイプ
- 文字ピッチ
- 行ピッチ
- ページ長
- ページ幅
- 印刷ページ数
- 文字セット
- フォーム名
- 部数

LP 印刷サービスは、これらの特性にデフォルト設定を提供します。ただし、印刷フィルタの方が一部の特性を効率よく処理できます。たとえば、プリンタによっては、LP 印刷サービスより効率よく複数のコピーを処理できるものがあり、その場合は複数コピーのページ制御用フィルタを提供できます。

プリンタ障害の検出

各プリンタは、独自の 방법으로プリンタ障害を検出し、障害信号を LP 印刷サービスに送信します。LP 印刷サービスは、印刷中にハングアップ (キャリア喪失) と過剰な遅延の有無をチェックするだけです。

プリンタによっては、優れた障害処理能力を持ち、障害の原因を記述するメッセージを送信できるものがあります。また、キャリア信号の喪失やデータフローのシャットオフを示すものとは異なる信号を使用して障害を示すプリンタもあります。これらの付加的なプリンタ障害情報を解釈するには、フィルタが必要です。

また、フィルタは印刷要求を保留し、プリンタ障害がクリアされるまで待ってから印刷を再開します。この機能を使用すると、中断された印刷要求全体を印刷し直す必要がありません。プリンタに使用される制御シーケンスを認識するフィルタだけが、ファイルの改ページ位置を判別できます。したがって、その種のフィルタ以外は、障害がクリアされた後で印刷を再開する必要があるファイル内の位置を検出できません。

印刷フィルタがメッセージを生成すると、そのメッセージは LP 印刷サービスによって処理され、警告が使用可能になっている場合は、システム管理者に警告が送信されます。87 ページの「障害の通知の設定」を参照してください。

印刷フィルタプログラムの要件

印刷フィルタは単純なものでも複雑なものでもかまいませんが、次の要件を満たさなければなりません。

- フィルタは、標準入力からファイルの内容を取得して、変換後のファイルを標準出力に送信しなければなりません。
- 外部ファイルを参照するプログラムはフィルタとして使用できません。一般にユーザーは `troff`、`nroff` などのワープロプログラムをフィルタとして使用しがちです。LP 印刷サービスは、フィルタプログラムから「組み込みファイル」と呼ばれる他のファイルへの参照を認識しません。`troff` と `nroff` はファイルを組み込むので、フィルタとして使用すると失敗することがあります。プログラムが処理を完了するために他のファイルを必要とする場合は、フィルタとして使用しないでください。
- フィルタは、一般にユーザーがアクセスできないファイルに依存しないでください。ユーザーが直接実行するときフィルタが失敗すると、LP 印刷サービスで実行しても失敗します。
- 低速フィルタは、ファイル内のエラーに関するメッセージを標準エラーに送信しますが、高速フィルタは送信しません。低速フィルタからのエラーメッセージは収集され、印刷要求を出したユーザーに送信されます。
- 低速フィルタが信号を受信したために終了すると、印刷要求が停止され、その要求を出したユーザーに通知されます。同様に、低速フィルタが 0 以外の終了コードを返して終了すると、印刷要求が停止され、ユーザーに通知されます。高速フィルタからの終了コードは、異なる方法で処理されます。

フィルタにプリンタ障害を検出させたい場合は、次の要件も満たしてください。

- フィルタは終了する前にできるだけ障害がクリアされるまで待つ必要があります。また、障害がクリアされたら、印刷を停止したページの先頭から印刷を再開してください。継続機能を使用したくない場合は、LP 印刷サービスは管理者に警告する前にフィルタを停止します。

- フィルタは、障害が認識されたらすぐにプリンタ障害メッセージを標準エラーに送信する必要があります。終了する必要はなく、障害がクリアされるまで待つことができます。
- フィルタは、ファイル内のエラーに関するメッセージを標準エラーに送信してはいけません。これらのメッセージは、ユーザーが読み込める標準出力に含める必要があります。
- フィルタは、ファイルの印刷が終了すると (ファイル内のエラーによって正常に印刷できなかった場合も) 終了コード 0 を返して終了する必要があります。
- フィルタは、プリンタ障害によって印刷要求を終了できなかった場合にのみ、0 以外の終了コードを返して終了する必要があります。
- フィルタをフィルタテーブルに追加する場合は、高速フィルタとして追加しなければなりません。

印刷フィルタ定義の作成

印刷フィルタ定義は、フィルタ、実行する印刷フィルタプログラム、実行する変換の種類などを LP 印刷サービスに指示します。/etc/lp/fd ディレクトリには、一連のフィルタ記述子ファイルが組み込まれています。これらのファイルは、フィルタの特性 (高速または低速フィルタなど) を記述し、フィルタプログラム (/usr/lib/lp/postscript/postdaisy など) を指します。

新しい印刷フィルタを定義するときには、フィルタプログラムを作成するだけでなく、印刷フィルタ定義を作成しなければなりません。印刷フィルタ定義には、LP 印刷サービスが使用する次の情報が入っています。

- 実行するフィルタプログラム名
- 受け付ける入力タイプ
- 生成する出力タイプ
- ジョブを送信できるプリンタタイプ
- ジョブを送信できる特定のプリンタ名
- フィルタタイプ (高速または低速)
- オプション

lpfilter コマンドへの直接入力として特性を入力できます。また、フィルタの特性を指定するファイルを作成し、ファイル名を lpfilter コマンドへの入力として使用することもできます。この種のファイルは「フィルタ記述子ファイル」と呼ばれ、/etc/lp/fd ディレクトリに配置してください。これらのファイルはフィルタそのものではなく、フィルタを指すものです。

情報をファイルに格納するかコマンド行に直接入力するかに関係なく、次の形式を使用してください。

```
Command: command-pathname [options]
Input types: input-type-list
Output types: output-type-list
Printer types: printer-type-list
Printers: printer-list
```

Filter type: fast または slow
Options: *template-list*

注 - Options 以外のフィルタ特性に複数の定義 (つまり複数行) を与えると、2 番目の定義のみが印刷サービスに使用されます。

情報はどんな順序で配置してもかまいません。また、すべての情報が必要とは限りません。値を指定しなければ、表 7-5 の値がデフォルトで割り当てられます。デフォルト値はあまり便利ではないので、明示的な値を指定するようにしてください。

表 7-5 lpfilter 引数のデフォルト値

項目	デフォルト
入力タイプ	任意
出力タイプ	任意
プリンタタイプ	任意
プリンタ	任意
フィルタタイプ	slow

コマンド

フィルタプログラムのフルパスを使用してください。プログラムが必ず必要とする固定オプションがある場合は、それらをこのコマンドに含めます。

入力タイプ

印刷フィルタが処理できるファイル内容形式のリストです。LP 印刷サービスには入力タイプ数の制限はありませんが、ほとんどのフィルタは 1 つのタイプしか受け付けません。いくつかのファイルタイプは、フィルタで処理できる程度に類似しています。14 文字以内の英数字とダッシュを使用した名前を指定できます。入力タイプ名の一部として下線 () は使用できません。

LP 印刷サービスは、一貫した命名規則に合致するように、これらの名前を使用してフィルタをファイルタイプと一致させます。たとえば、複数のフィルタが同じ入力タイプを受け付ける場合は、各フィルタ用に指定するときに、その入力タイプに同じ名前を使用します。ユーザーには、ファイルの印刷を依頼するときにファイルタイプの識別方法がわかるように、これらの名前を通知してください。

出力タイプ

出力タイプは、フィルタが出力として生成できるファイルタイプのリストです。入力タイプごとに、フィルタは1つずつ出力タイプを生成します。ただし、出力タイプはジョブごとに異なることがあります。出力タイプ名は、14文字以内の英数字とダッシュ(-)です。

出力タイプ名は、利用できる(ローカルまたはリモート)プリンタのタイプと一致するか、他のフィルタで処理される入力タイプと一致しなければなりません。LP印刷サービスは、ファイルを変換するために異なるフィルタを複数回かける必要があることがわかると、フィルタをシェルパイプラインにグループ化します。このように複雑な処理はほとんど必要ありませんが、LP印刷サービスではこの処理を実行できます。ユーザーが印刷したいあらゆるファイルを入力タイプにするフィルタの集合を探して、各ファイルをプリンタで処理できるファイルタイプに直接変換してください。

プリンタタイプ

プリンタタイプは、印刷フィルタがファイルを変換できるプリンタタイプのリストです。ほとんどのプリンタとフィルタの場合、これは出力タイプのリストと同じなので、フィルタ定義のこの部分はブランクにしておいてかまいません。しかし、別のタイプを使用することもできます。たとえば、初期化には1つのプリンタタイプを使用するプリンタでも、複数のファイル内容形式を認識できるものがあります。実際には、このプリンタには、各種ファイルタイプを、処理できるファイルタイプに変換する内部フィルタが付いています。したがって、1つのフィルタは、プリンタで処理できるファイルタイプに合った複数の出力タイプのいずれかを生成できます。印刷フィルタには、そのプリンタタイプで機能することを示すマークを付ける必要があります。

もう1つの例として、同じファイルタイプを受け付けるものとして表示される2つのモデルのプリンタを使用できます。ただし、内部に若干違いがあるので、一方のプリンタが生成する結果と異なります。それらのプリンタには、AとBなど、異なるプリンタタイプであることを示すラベルを付けます。この場合、Bはズレがある方のプリンタです。タイプBのプリンタで生成されるズレを考慮してファイルを調整するフィルタを作成します。このフィルタはこの種のプリンタタイプにのみ必要なもので、タイプBプリンタでのみ機能するものとして指定します。

プリンタ

一般に、印刷フィルタはその出力を受け付けるどのプリンタでも機能するので、通常はフィルタ定義のこの部分をスキップできます。

ただし、プリンタによっては、フィルタが生成する出力に適したものと適さないものがあります。たとえば、1台のプリンタを高速出力専用にして、フィルタを通す必要がないファイルのみをそのプリンタに送信できます。同じタイプの他のプリンタは、印刷前に広範囲にフィルタを通す必要があるファイルに使用できます。

フィルタタイプ

LP 印刷サービスは、144 ページの「フィルタのタイプ」で説明しているように、高速フィルタと低速フィルタを認識します。

印刷モードで (lp -y コマンドを使用して) 呼び出される低速フィルタは、印刷要求が出されたシステム上で実行しなければなりません。LP 印刷サービスはモード値をプリンタサーバーに渡せません。ただし、ファイル内容形式 (lp コマンドの -T オプションの後で指定) をプリンタサーバー上の内容形式と一致させることはできます。したがって、プリンタサーバー上で特殊モードを有効にしたい場合は、LP 印刷サービスが入力タイプと出力タイプを一致することができる内容形式を指定しなければなりません。

オプション

オプションは、各種の情報をフィルタコマンドのコマンド行引数にどのように変換するかを指定します。この情報には、ユーザーからの (印刷要求を伴う) 指定、プリンタ定義、要求の処理に使用されるフィルタによって実装される仕様などを含めることができます。

テンプレートによる印刷フィルタオプションの定義

印刷フィルタオプションの定義には 13 個の情報源があり、それぞれが「キーワード」で表されています。各オプションは「テンプレート」内で定義されます。テンプレートとは、フィルタのいずれかの特性値に基づいてフィルタコマンドに渡されるオプションを定義する、フィルタ定義内のステートメントです。

フィルタ定義で指定するオプションには、13 個のキーワードを使用しなくても、すべて使用しても、そのサブセットを使用してもかまいません。また、完全なフィルタ定義に複数の定義が必要な場合は、1 つのキーワードを複数回指定することもできます。印刷フィルタ定義における options を定義するための 13 個のキーワードについては、表 7-6 を参照してください。

表 7-6 印刷フィルタオプションのキーワード

特性	キーワード	考えられるパターン	例
内容形式 (入力)	INPUT	内容形式	troff
内容形式 (出力)	OUTPUT	内容形式	postscript, impress
プリンタタイプ	TERM	プリンタタイプ	att495
プリンタ名	PRINTER	プリンタ名	lp1

表 7-6 印刷フィルタオプションのキーワード (続き)

特性	キーワード	考えられるパターン	例
文字ピッチ	CPI	四捨五入された 10 進数	10
行ピッチ	LPI	四捨五入された 10 進数	6
ページ長	LENGTH	四捨五入された 10 進数	66
ページ幅	WIDTH	四捨五入された 10 進数	80
印刷ページ数	PAGES	ページリスト	1-5,13-20
文字セット	CHARSET	文字セット	finnish
フォーム名	FORM	フォーム名	invoice2
部数	COPIES	整数	3
特殊モード	MODES	モード	landscape

印刷フィルタ定義には、複数のテンプレートを含めることができます。複数のテンプレートを指定する場合は、1 行にコンマで区切って入力するか、先頭に `Options:` を付けて複数行に入力します。

テンプレートの形式は次のとおりです。

keywordpattern = replacement

`keyword` は、フィルタの特定の特性に関して登録されたオプションのタイプを識別します。

`pattern` は、キーワードの特定のオプションです。

`replacement` は、キーワードが指定した値のときに発生する動作です。

特定のフィルタのオプションを定義する方法を示す例として、印刷サービススケジューラを使って次の条件でフィルタに印刷要求を割り当てたいものと仮定します。

- フィルタで生成される OUTPUT のタイプが `impress` の場合は、フィルタに `-I` オプションを渡す。
- フィルタで生成される OUTPUT のタイプが `postscript` の場合は、フィルタに `-P` オプションを渡す。

上記の条件を指定するには、`lpfilter` コマンドのオプションとして次のテンプレートを与えます。

`Options: OUTPUT impress=-I, OUTPUT postscript=-P`

Options 行が長くなりすぎる場合は、次のように各テンプレートを別々の行に入れません。

```
Options: OUTPUT impress=-I
Options: OUTPUT postscript=-P
```

どちらのテンプレートでも、*keyword* は OUTPUT として定義されています。第 1 のテンプレートでは、パターンは *impress* で、*replacement* の値は *-I* です。第 2 のテンプレートでは、*pattern* の値は *postscript* で、*replacement* の値は *-P* です。

各種テンプレート (つまり、各キーワードの *pattern* 引数と *replacement* 引数) に与える値を見つけるには、次のことに注意する必要があります。

- INPUT テンプレートの値は、フィルタによる変換が必要なファイル内容形式からとっています。
- OUTPUT テンプレートの値は、フィルタが生成すべき出力タイプからとっています。
- TERM テンプレートの値はプリンタタイプです。
- PRINTER テンプレートの値は、最終出力を印刷するプリンタ名です。
- CPI、LPI、LENGTH、WIDTH の各テンプレートの値は、ユーザーの印刷要求、使用するフォーム、またはプリンタのデフォルト値からとっています。
- PAGES テンプレートの値は、印刷すべきページのリストです。通常、これはコマンドで区切ったページ範囲のリストです。各ページ範囲は、ダッシュで区切った 1 対の数値、または単一の数字からなります (たとえば、1-5,6,8,10 は 1 ページから 5 ページまでと、6 ページ、8 ページ、10 ページを示します)。ただし、印刷要求に *-P* オプションで値を与えると、印刷要求は変更されずに渡されます。
- CHARSET テンプレートの値は、使用する文字セット名です。
- FORM テンプレートの値は、*lp -f* コマンド (印刷要求を出すのに使用するコマンド) で要求したフォーム名です。
- COPIES テンプレートの値は、ファイルの印刷部数です。フィルタがこのテンプレートを使用する場合、この「1 部」にはフィルタによって生成される複数のコピーが含まれるので、LP 印刷サービスはフィルタがかけられたファイルの印刷部数から 1 を引きます。
- MODES テンプレートの値は、*lp -y* コマンドからとっています。ユーザーは複数の *-y* オプションを指定できるので、MODES テンプレートは複数の値をとることができます。値はユーザーが与えた順に左から右に適用されます。

テンプレートの *replacement* 部は、テンプレートの値をフィルタプログラムに与える方法を示します。通常はリテラルオプションですが、値の位置を示すプレースホルダのアスタリスク (*) が付いていることがあります。*pattern* と *replacement* には、ユーザー入力オプションからフィルタオプションへの複雑な変換を行うために、*ed* コマンドの正規表現の構文を使用することもできます。*ed* の正規表現の構文はすべて使用できます。たとえば、*\(...\)* や *\n* 構成を使用すると、*pattern* の各部を抽出して *replacement* にコピーできます。また、*&* を使用すると、*pattern* 全体を *replacement* にコピーできます。詳細は、*ed* (1) のマニュアルページを参照してください。

注 - *pattern* または *replacement* にコンマまたは等号 (=) を含める場合は、その前にバックスラッシュ (\) を付けてください。これらの文字の前にバックスラッシュが付いていると、*pattern* または *replacement* が使用されるときに削除されます。

▼ 新しい印刷フィルタを作成する方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. 印刷フィルタプログラムを作成します。
印刷フィルタプログラムの概要については、144 ページの「印刷フィルタプログラムの作成」を参照してください。使用しやすいように、通常、フィルタ定義は /usr/lib/lp/postscript ディレクトリに入っています。作成したプログラムは、選択したディレクトリ内の /usr/lib/lp ディレクトリの下に入れる必要があります。
3. 印刷フィルタ定義を作成します。
印刷フィルタ定義の概要については、147 ページの「印刷フィルタ定義の作成」を参照してください。印刷フィルタ定義はテキストファイルに保存する必要があります。使用しやすいように、通常、フィルタ定義は /etc/lp/fd ディレクトリに入っており、接尾辞 .fd で識別されます。
4. 印刷フィルタをプリンタサーバーに追加します。
詳細は、116 ページの「印刷フィルタを追加する方法」を参照してください。

例 — 新しい印刷フィルタを作成する

次の例は、N37 または Nlp を simple に変換する印刷フィルタ定義を示します。

```
Input types: N37, Nlp, simple
Output types: simple
Command: /usr/bin/col
Options: MODES expand = -x
Options: INPUT simple = -p -f
```

次の例で、印刷フィルタプログラム名は col です。新しい印刷フィルタをプリンタサーバーに追加すると、ユーザーの印刷要求は次のように処理されます。

- ユーザーが次のコマンドを入力した場合

```
$ lp -y expand report.doc
```

印刷プログラムは次の引数を使用して実行され、ファイルが変換されます。

```
/usr/bin/col -x -p -f
```

- ユーザーが次のコマンドを入力した場合

```
$ lp -T N37 -y expand report.doc
```

印刷プログラムは次の引数を使用して実行され、ファイルが変換されます。

```
/usr/bin/col -x
```

次の例は、troff から PostScript に変換する印刷フィルタ定義を示します。

```
Input types: troff
```

```
Output types: postscript
```

```
Printer types: PS
```

```
Filter type: slow
```

```
Command: /usr/lib/lp/postscript/dpost
```

```
Options: LENGTH * = -l*
```

```
Options: MODES port = -pp, MODES land = -pl
```

```
Options: MODES group \=\([1-9]\) = -n\1
```

次の例で、フィルタプログラム名は dpost です。このプログラムは入力タイプ troff をとり、postscript 出力を生成し、タイプ PS (PostScript) のプリンタに機能します。ユーザーは、用紙方向を縦モードにするか横モードにするかを尋ねるプロンプトが表示されたときに、それぞれの省略形 port または land を指定するだけですみます。これらのオプションは LP 印刷サービスに固有ではないので、ユーザーは lp -y コマンドを使用して指定しなければなりません。

新しい印刷フィルタをプリンタサーバーに追加すると、印刷要求は次のように処理されます。

- ユーザーが次のコマンドを入力し、横方向、ページ長 60 行で、troff ファイルタイプを PostScript プリンタ (タイプ PS) で印刷するように要求した場合

```
$ lp -T troff -o length=60 -y land -d luna ch1.doc
```

次の引数が指定された印刷フィルタプログラム dpost コマンドは、ファイルを変換します。

```
/usr/lib/lp/postscript/dpost -l60 -pl luna ch1.doc
```

- ユーザーが次のコマンドを入力した場合

```
$ lp -T troff -y group=4 -d luna ch1.doc
```

次の引数が指定された印刷フィルタプログラム dpost コマンドは、ファイルを変換します。

```
/usr/lib/lp/postscript/dpost -n4
```

新しいプリンタフォームの作成

新しいフォームを提供したい場合は、lpforms コマンドへの入力として9個の必須特性 (ページ長とページ幅など) に関する情報を入力し、その特性を定義しなければなりません。LP 印刷サービスは、この情報を次の2つの目的に使用します。

- フォーム上に正しく印刷されるようにプリンタを初期化する。
- フォームの処理方法に関する留意事項をシステム管理者に送信する。

フォーム名は、14文字以内の英数字と下線であれば、任意に選択して使用できます。情報は次の形式でなければなりません。

```
Page length: scaled number
Page width: scaled number
Number of pages: integer
Line pitch: scaled number
Character pitch: scaled number
Character set choice: character-set-name [,mandatory]
Ribbon color: ribbon-color
Comment:
informal notes about the form
Alignment pattern: [content-type] alignment pattern
```

省略可能な句、[,mandatory] は、ユーザーがフォームの文字セット選択を無効にできないことを意味します。*content-type* は位置揃えパターンを使用して指定できますが省略可能です。この属性を指定すると、印刷サービスは必要に応じて使用し、ファイルにフィルタをかけて印刷する方法を決定します。

2つの例外がありますが、情報は任意の順序で指定できます。例外は、Alignment pattern (位置揃えパターン) (必ず最後に配置しなければなりません) と comment (コメント) (必ず Comment: プロンプトの行に続かなければなりません) です。コメントにキー句 (Page length、Page width など) で始まる行が含まれている場合は、キー句が行頭にならないように、その行を> 文字で始めます。先頭の> 文字は、コメントから除去されて表示されません。

すべての情報を与えなければならないわけではありません。表 7-7 の項目の値を指定しないときは、デフォルト値が割り当てられます。lpforms コマンドを実行する前に、新しいフォームに関して次の情報を収集してください。

表 7-7 フォームのデフォルト値

項目	デフォルト	説明
ページ長	66 行	フォームの長さ、または複数ページのフォームの場合は各ページの長さ。この情報は、行数でもインチ単位やセンチメートル単位でもかまわない

表 7-7 フォームのデフォルト値 (続き)

項目	デフォルト	説明
ページ幅	80 列	文字数、インチ数、またはセンチメートル数によるフォームの幅
ページ数	1	複数ページのフォームのページ数。LP 印刷サービスは、この数値と印刷フィルタ (利用できる場合) を使用して、位置揃えパターンを 1 つのフォームの長さに制限する。「位置揃えパターン」の説明を参照。フィルタが利用できない場合は、LP 印刷サービスは出力を打ち切らない
行ピッチ	1 インチあたり 6 行	フォーム上の行間隔。これは「リーディング」とも呼ばれる。2 行間の間隔、つまりベースラインからベースラインまでの間隔を 1 インチまたは 1 センチあたりの行数で表す
文字ピッチ	1 インチあたり 10 文字	フォームに表示される文字間隔。文字の間隔を 1 インチまたは 1 センチあたりの文字数で表す
文字セット選択肢	任意	このフォームに使用しなければならない文字セット、印字ホイール、またはフォントカートリッジ。ユーザーは、このフォームを使用するときに独自の印刷要求に別の文字セットを選択できる。また、単一の文字セットのみを使用するように指示できる
リボンの色	任意	フォームを常に特定のカラーリボンで印刷しなければならない場合、LP 印刷サービスはどの色を使用すべきかを示す装着警告メッセージを表示できる
コメント	(デフォルトなし)	ユーザーがフォームを理解する上で参考になる任意の情報。たとえば、フォーム名、そのバージョン、用途、または使用上の制限を示すことができる
位置揃えパターン	(デフォルトなし)	LP 印刷サービスが 1 枚のブランクフォームを埋めるために使用するサンプルファイル。フォームを装着するときに、このパターンを印刷して正しく位置揃えすることができる。また、印刷サービスに印刷方法が認識されるように、このパターンの内容形式を定義することもできる

注 - LP 印刷サービスは、位置揃えパターン内の重要な情報にマスクをかけようとしません。小切手を位置揃えするときなど、サンプルフォームに重要な情報を印刷したくない場合は、該当するデータにマスクをかける必要があります。LP 印刷サービスは、スーパーユーザーまたは lp としてログインしたユーザー以外は読み取れないように、位置揃えパターンを安全な場所に格納します。

フォーム情報を収集し終わったら、フォームを lpforms コマンドに入力します。lpforms コマンドに入力する前にこの情報を編集できるように、まず、この情報を別のファイルに記録してください。そうすれば、プロンプトの後で個々の情報を入力しなくても、そのファイルを入力として使用できます。

▼ 新しいフォーム定義を作成する方法

1. プリンタサーバーにスーパーユーザーまたは lp としてログインするか、同等の役割になります。
2. フォーム定義ファイルを作成します。
印刷フォームの作成方法の概要については、155 ページの「新しいプリンタフォームの作成」を参照してください。プリンタ定義はテキストファイルに保存してください。
3. フォームを LP 印刷サービスに追加します。

```
# lpadmin -p printer-name -M -f form-name
```
4. フォームをプリンタサーバーに追加します。
手順については、120 ページの「フォームを追加する方法」を参照してください。

第 8 章

LP 印刷サービス (リファレンス)

この章では、LP 印刷サービスの内容説明を提供します。

- 160 ページの「LP 印刷サービスの構造」
- 168 ページの「LP 印刷サービスのコマンド」
- 169 ページの「LP 印刷サービスの機能」
- 169 ページの「LP によるファイルの管理とローカル印刷要求のスケジューリングの方法」
- 170 ページの「ネットワーク印刷要求のスケジューリング」
- 171 ページの「印刷ファイルにフィルタを適用する」
- 171 ページの「プリンタインタフェースプログラムの機能」
- 171 ページの「lpsched デーモンによる印刷ジョブ状態の確認」
- 172 ページの「ログファイルの消去」

プリンタ関連作業の情報の参照箇所

プリンタ関連作業	参照箇所
Solaris プリンタマネージャと lp コマンドによるプリンタのセットアップ	第 4 章
セットアップ後の lp コマンドによるプリンタの管理	第 5 章
文字セット、フィルタ、フォーム、フォントの管理	第 6 章
LP 印刷サービスのカスタマイズ	第 7 章

LP 印刷サービス

「LP 印刷サービス」とは、ユーザーが作業を続けながらファイルを印刷できるようにするソフトウェアユーティリティの集合です。当初、印刷サービスは LP スプーラと呼ばれていました (LP はラインプリンタの意味ですが、現在ではレーザープリンタなどのさまざまな種類のプリンタも含まれます。スプール (Spool) は、system peripheral operation off-line の頭文字です)。

印刷サービスは、LP 印刷サービスソフトウェア、システム管理者が提供する印刷フィルタ、ハードウェア (プリンタ、システム、およびネットワーク接続) からなっています。

LP 印刷サービスの構造

この節では、LP 印刷サービスのディレクトリ構造、ファイル、ログ、およびコマンドについて説明します。

LP 印刷サービスのディレクトリ

LP 印刷サービスのファイルは、表 8-1 のように 7 つのディレクトリに分散されています。

表 8-1 LP 印刷サービスのディレクトリ

ディレクトリ	内容
/usr/bin	LP 印刷サービスのユーザーコマンド
/etc/lp	LP 構成ファイルの階層
/usr/share/lib	terminfo データベースディレクトリ
/usr/lib/print	lp 変換スクリプト、in.lpd デーモン、printd デーモン。 printd デーモンは /var/spool/print ディレクトリのすべての保留ジョブを 1 分間に 1 回転送する。転送するジョブがなくなると終了する
/usr/sbin	LP 印刷サービスの管理コマンド
/usr/lib/lp	lpsched プログラム、バイナリファイル、PostScript フィルタ、および model ディレクトリ。model ディレクトリには、標準プリンタインタフェースプログラムが入っている

表 8-1 LP 印刷サービスのディレクトリ (続き)

ディレクトリ	内容
/var/lp/logs	LP ログファイル。lp sched .n には lp sched からのメッセージが、requests .n には完了した印刷要求に関する情報が含まれる
/var/spool/lp	ファイルが印刷待ち行列に入れられるスプーリングディレクトリ
/var/spool/print	LP 印刷サービスのクライアント側要求格納域

LP 印刷サービスの構成ファイル

lp sched デーモンが/etc/lp ディレクトリに格納する構成情報を、次の表に示します。



注意 – 表 8-2 の構成ファイルはプライベートインタフェースで、将来のリリースでは変更される可能性があります。現在の場所にあるファイルに依存する、または現在使用している形式のデータに依存するソフトウェアを構築しないようにしてください。

表 8-2 /etc/lp ディレクトリの内容

ファイル	種類	説明
classes	ディレクトリ	lpadmin -c コマンドで与えたクラスを識別するファイル
fd	ディレクトリ	既存のフィルタの記述
filter.table	ファイル	印刷フィルタ照合テーブル
forms	ディレクトリ	各フォームのファイルを格納する場所。最初は、このディレクトリは空になっている
interfaces	ディレクトリ	プリンタインタフェースプログラムファイル
logs	/var/lp/logs へのリンク	印刷動作のログファイル
model	/usr/lib/lp/model へのリンク	標準プリンタインタフェースプログラムのリンク
printers	ディレクトリ	各ローカルプリンタのディレクトリ。各ディレクトリには、個々のプリンタの構成情報と警告ファイルが入っている

表 8-2 /etc/lp ディレクトリの内容 (続き)

ファイル	種類	説明
pwheels	ディレクトリ	印字ホイールまたはカートリッジファイル

これらの構成ファイルは、lpd ベースのプリンタサーバー上の /etc/printcap ファイルと同じ機能を提供します。

注 - これらのファイルの内容を確認できますが、ファイルを直接編集しないでください。代わりに、lpadmin コマンドを使用して構成を変更します。変更内容は /etc/lp ディレクトリ内の構成ファイルに入力されます。lpsched デーモンは、構成ファイルを管理して構成します。

/etc/lp/printers ディレクトリには、システムに認識されるローカルプリンタごとに1つずつサブディレクトリが入っています。次の例は、プリンタ sparcl1 と luna の /etc/lp/printers サブディレクトリを示します。

```
$ ls -l /etc/lp/printers
drwxrwxr-x 2 lp lp 512 Jan 23 23:53 luna
drwxrwxr-x 2 lp lp 512 Jan 11 17:50 sparcl1
```

プリンタ固有の各ディレクトリ内では、次のファイルを使用してプリンタを記述できます。

ファイル名	説明
alert.sh	警告に応答して実行するシェル
alert.vars	警告変数
configuration	構成ファイル
users.deny	プリンタアクセスが拒否されるユーザーのリスト
comment	プリンタ記述

通常、プリンタ luna の構成ファイル /etc/lp/printers/luna/configuration は、次のようになっています。

```
Banner: on: Always
Content types: PS
Device: /dev/term/b
Interface: /usr/lib/lp/model/standard
Printer type: PS
Modules: default
```

terminfo データベース

/usr/share/lib ディレクトリには、terminfo データベースのディレクトリが入っており、そのディレクトリには多数のタイプの端末とプリンタに関する定義が入っています。LP 印刷サービスは、terminfo データベース内の情報を使用してプリンタを初期設定し、選択されたページサイズ、文字ピッチ、行ピッチ、および文字セットを設定し、一連のコードをプリンタに送ります。

各プリンタは、terminfo データベース内で短縮名を使用して識別されます。terminfo データベースの構造については、48 ページの「プリンタタイプ」を参照してください。必要であれば、terminfo データベースにエントリを追加できますが、これは煩雑で時間のかかる作業です。詳細は、135 ページの「サポートされていないプリンタの terminfo エントリを追加する」を参照してください。

デーモンと LP 内部ファイル

/usr/lib/lp ディレクトリには、表 8-3 に示すような LP 印刷サービスに使用されるデーモンとファイルが入っています。

表 8-3 /usr/lib/lp ディレクトリの内容

ファイル	種類	説明
bin	ディレクトリ	印刷警告、低速フィルタ、待ち行列管理プログラムを生成するファイルが入っている
lpsched	デーモン	LP 印刷要求のスケジューリングを管理する
model	ディレクトリ	標準プリンタインタフェースプログラムが入っている
postscript	ディレクトリ	LP 印刷サービスで提供されているすべての PostScript フィルタプログラムが入っている。これらのフィルタには、フィルタの特性とその格納場所を LP 印刷サービスに指示する /etc/lp/fd ディレクトリ内の記述子ファイルが含まれる

LP 印刷サービスのログファイル

LP 印刷サービスは、次の 2 組のログファイルを管理します。

ログファイル名	説明
syslogd (1M)	/etc/syslog.conf の lpr.debug を設定して LP 印刷サービスロギングを有効にする

ログファイル名	説明
<code>/var/spool/lp</code>	印刷待ち行列に入っている現在の待ち行列のリスト
<code>/var/lp/logs/requests</code>	進行中の印刷要求の履歴

印刷待ち行列ログ

各システムのスケジューラは、`/var/spool/lp/tmp/system` ディレクトリと `/var/spool/lp/requests/system` ディレクトリ内で印刷要求ログを保管します。各印刷要求は、要求に関する情報を含んだファイルを2つ(各ディレクトリに1つずつ)持っています。`/var/spool/lp/requests/system` ディレクトリ内の情報には、スーパーユーザーまたは `lp` しかアクセスできません。`/var/spool/lp/tmp/system` ディレクトリ内の情報には、その要求を出したユーザー、スーパーユーザー、または `lp` しかアクセスできません。

次の例では、`/var/spool/lp/tmp/starbug` ディレクトリの内容を示します。

```
$ ls /var/spool/lp/tmp/starbug
5      5-0
# cat 5-0
C 1
D print1
F /etc/profile
P 20
T /etc/profile
t simple
U root
s 0000
v 2
```

これらのファイルは、印刷要求が待ち行列に入っている限り、そのディレクトリ内に残っています。要求が完了すると、ファイル内の情報は組み合わせられ、`/var/lp/logs/requests` ファイルに追加されます。このファイルについては、次の節で説明します。

現在待ち行列に入っている印刷要求の状態を追跡したい場合は、`/var/spool/lp/logs` ディレクトリの情報を使用します。

履歴ログ

LP 印刷サービスは、`lpsched` と `requests` という2つのログファイルに印刷サービスの履歴を記録します。これらのログファイルは、`/var/lp/logs` ディレクトリに入っています。これらのログ内の情報を使用し、印刷の問題を診断して解決できます。次の例は、`/var/lp/logs` ディレクトリの内容を示します。

```
# cd /var/lp/logs
# ls
lpsched.1      requests      requests.2
```

```
lp sched      lp sched.2    requests.1
#
```

問題の解決に最も重要な2つのログファイルは、ローカルの印刷要求に関する情報が入っている `lp sched` ログと、すでに完了してプリンタの待ち行列にない印刷要求の情報が入っている `requests` ログです。

`requests` ログの構造は単純なので、共通の UNIX シェルコマンドを使用してデータを抽出できます。要求は、出力された順番に要求 ID を示す行で区切って表示されます。= で始まる区切り行より下の各行には、その行に含まれる情報の種類を識別する1つの文字が付いています。各文字は、空白文字1つでデータと区切られます。

次の例は、`requests` ログの内容を示しています。

```
# pwd
/var/lp/logs
# tail requests.2
= print1-3, uid 0, gid 1, size 206662, Wed Nov 14 08:56:30 MST 2001
z print1
C 1
D print1
F /usr/dict/words
P 20
T /usr/dict/words
t simple
U root
s 0x0014
v 2
#
```

表 8-4 に、文字コードとそれに対応する LP `requests` ログ内の行を示します。

表 8-4 LP `requests` ログ内の文字コード

文字	行の内容
=	区切り行。ユーザーの要求 ID、ユーザー ID (UID)、グループ ID (GID)、フィルタを通す前のファイルサイズの総バイト数、および要求が待ち行列に入れられた時刻が入っている
z	プリンタ名
C	印刷部数
D	出力先のプリンタクラス、または any
F	印刷されたファイル名。この行は印刷されたファイルごとに区切られ、ファイルが印刷された順番で表示される
f	(省略可能) 使用するフォーム名。
H	(省略可能) 特殊処理。resume、hold、または immediate

表 8-4 LP requests ログ内の文字コード (続き)

文字	行の内容
N	(省略可能) 印刷要求が正常に完了したときに使用された警告のタイプ。 ユーザーが電子メールで通知を受けたときは m タイプ、端末へのメッセージで通知を受けた場合は w タイプ
O	(省略可能) プリンタ依存の -o オプション。たとえば、nobanner
P	印刷要求の優先順位
p	印刷されたページのリスト
r	(省略可能) ユーザーが lp -r コマンドを使用してファイルの「raw」処理を要求したときに含まれる 1 文字の行
S	(省略可能) 使用された文字セット、印字ホイール、またはカートリッジ
T	バナーページに印刷されるタイトル
t	ファイル内で見つかった内容形式
U	印刷要求を出したユーザー名
s	16 進数形式の各ビットの組み合わせで表される要求の結果。印刷サービスの内部では複数のビットが使用される。ビットとその意味については、下表に記載
x	(省略可能) 印刷要求に使用された低速フィルタ
Y	(省略可能) 要求の印刷に使用された印刷フィルタの特殊モードのリスト
z	(省略可能) 要求に使用したプリンタ。要求がプリンタまたはプリンタクラスの待ち行列に入れられた場合や、要求が別の出力先に移動された場合は、このプリンタは出力先 (d 行) とは異なる

表 8-5 に、LP requests ログ内の結果コードとその説明を示します。

表 8-5 LP requests ログ内の結果コード

結果コード	説明
0x0001	要求は保留され再開を待機中
0x0002	低速フィルタを実行中
0x0004	低速フィルタを正常に完了
0x0008	要求はプリンタ上にある
0x0010	印刷を正常に完了
0x0020	要求は保留されユーザーによる変更を待機中
0x0040	要求は取り消し済み

表 8-5 LP requests ログ内の結果コード (続き)

結果コード	説明
0x0080	要求は次に印刷される
0x0100	フィルタ処理または印刷要求に失敗
0x0200	要求はリモートプリンタに転送中 (現在は使用されない)
0x0400	ユーザーに通知
0x0800	通知が動作中
0x1000	リモートシステムが要求を受け付け済み (現在は使用されない)
0x2000	管理者が要求を保留した
0x4000	プリンタのフィルタを変更しなければならなかった
0x8000	要求は一時的に停止された

スプーリングディレクトリ

印刷待ち行列に入れられたファイルは、印刷されるまで `/var/spool/lp` ディレクトリに格納されますが、それがわずかに数秒の場合があります。表 8-6 は、`/var/spool/lp` ディレクトリの内容を示します。

表 8-6 `/var/spool/lp` ディレクトリの内容

ファイル	種類	説明
SCHEDLOCK	ファイル	スケジューラのロックファイル。スケジューラが停止し、再起動されない場合は、このファイルをチェックする
admins	ディレクトリ	<code>/etc/lp</code> へのリンク
bin	ディレクトリ	<code>/usr/lib/lp/bin</code> へのリンク
logs	リンク	完了した印刷要求のログが記録される <code>../lp/logs</code> へのリンク
model	リンク	<code>/usr/lib/lp/model</code> へのリンク
requests	ディレクトリ	印刷要求が印刷されるまで記録される構成済みプリンタごとのサブディレクトリが入ったディレクトリ。ユーザーはこのログにアクセスできない
system	ディレクトリ	システムの印刷状態ファイル
temp	リンク	スプールされた要求が入っている <code>/var/spool/lp/tmp/hostname</code> へのリンク

表 8-6 /var/spool/lp ディレクトリの内容 (続き)

ファイル	種類	説明
tmp	ディレクトリ	印刷要求が印刷されるまでログが記録される構成済みの各プリンタのディレクトリ。既存の印刷要求を変更した場合も、このログに記録される

LP 印刷サービスのコマンド

表 8-7 に、頻繁に使用する LP 印刷サービスのコマンドを示します。(1M) のコマンドを使用するには、スーパーユーザーまたは lp になるか、同等の役割になる必要があります。

表 8-7 LP 印刷サービスコマンド早見表

コマンド	マニュアルページ	作業
enable	enable(1)	プリンタを使用可能にする
cancel	cancel(1)	印刷要求を取り消す
lp	lp(1)	1 つ以上のファイルをプリンタに送る
lpstat	lpstat(1)	LP 印刷サービスの状態を出力する
disable	enable(1)	1 台以上のプリンタを無効にする
accept	accept(1M)	印刷要求を特定の出力先の待ち行列に入れられるようにする
reject	accept(1M)	印刷要求が特定の出力先の待ち行列に入れられないようにする
lpadmin	lpadmin(1M)	プリンタの構成を設定または変更する
lpfilter	lpfilter(1M)	フィルタの定義を設定または変更する
lpforms	lpforms(1M)	あらかじめ印刷されたフォームを設定または変更する
lpadmin	lpadmin(1M)	フォームを取り付ける
lpmove	lpmove(1M)	ある出力先から別の出力先に出力要求を移動する
lpsched	lpsched(1M)	LP 印刷サービススケジューラを起動する
lpshut	lpshut(1M)	LP 印刷サービススケジューラを停止する
lpusers	lpusers(1M)	デフォルトの優先順位と、LP 印刷サービスのユーザーが要求できる優先順位の制限を設定または変更する

LP 印刷サービスの機能

LP 印刷サービスは、次の機能を実行します。

- ファイルを管理してローカル印刷要求をスケジュールする。
- ネットワーク要求を受け取り、スケジュールする。
- ファイルが正しく印刷されるように、必要に応じてフィルタを通す。
- プリンタとインタフェースするプログラムを起動する。
- ジョブの状態を追跡する。
- プリンタに取り付けられたフォームを追跡する。
- 現在装着されている印字ホイールを追跡する。
- 新しいフォームや別の印字ホイールを取り付け、装着するよう警告を発する。
- 印刷問題に関する警告を発する。

LP によるファイルの管理とローカル印刷要求のスケジュールリングの方法

LP 印刷サービスには、`lpsched` というスケジューラデーモンが組み込まれています。スケジューラデーモンは、プリンタの設定と構成に関する情報を使用して LP システムファイルを更新します。

また、`lpsched` デーモンは、ユーザーが要求をアプリケーションから出すかコマンド行から出すかに関係なく、図 8-1 のように、プリンタサーバー上のすべてのローカル印刷要求をスケジュールします。さらに、スケジューラはプリンタとフィルタの状態を追跡します。プリンタが要求を印刷し終わると、プリンタサーバー上の待ち行列に残っているものがあれば、スケジューラは次の要求をスケジュールします。

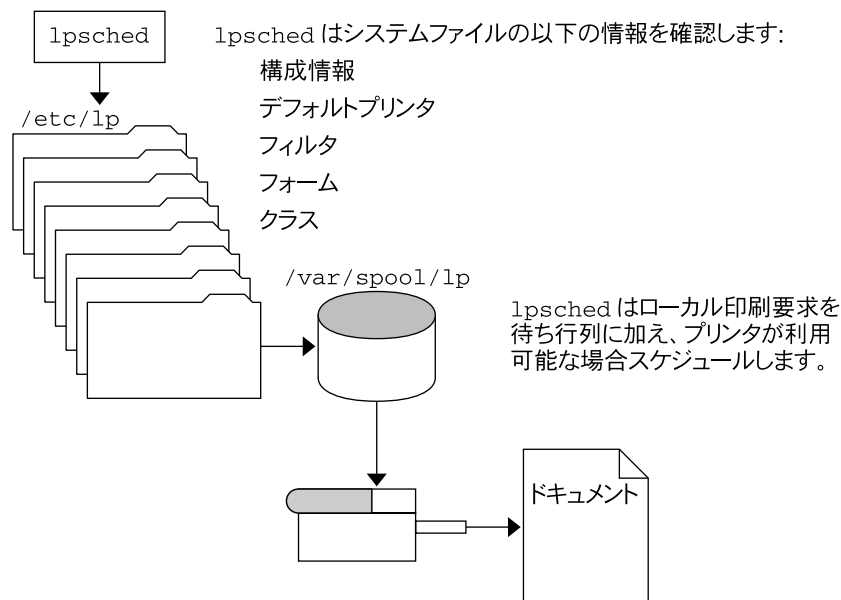


図 8-1 lpsched デーモンによるローカル印刷要求のスケジューリング方法

各プリンタサーバーは、LP スケジューラを 1 つだけ実行していなければなりません。スケジューラは、システムのブート時に (または実行レベル 2 を入力すると)、制御スクリプト `/etc/rc2.d/s80lp` によって起動されます。システムを再起動しなくても、`/etc/init.d/lp stop` コマンドを使用してスケジューラを停止し、`/etc/init.d/lp start` コマンドを使用して再起動できます。各システムのスケジューラは、`lp` コマンドによってシステムに出された要求を管理します。

ネットワーク印刷要求のスケジューリング

各印刷クライアントは、ネットワーク経由で直接、プリンタサーバーと通信します。通信は、要求コマンド (`lp`、`lpstat`、`cancel`、`lpr`、`lpq`、または `lprm`) とプリンタサーバー上の印刷サービス間で行われます。これによって、クライアント専用システムにおける印刷システムのオーバーヘッドが軽減され、その結果、拡張性、パフォーマンス、およびデータの正確性が向上します。

プリンタサーバーは、現在、インターネットサービスデーモン (`inetd`) で印刷要求を待機します。ネットワークから印刷サービスへ要求があると、`inetd` デーモンは「プロトコルアダプタ」と呼ばれるプログラム (`in.lpd`) を起動します。プロトコルアダプタは、印刷要求を翻訳して、印刷スプーラに送信し、その結果を要求依頼元に戻します。このプロトコルアダプタは、要求の発生時に起動し、ネットワーク要求の処理が完了すると終了します。このプロセスによって、印刷のためのアイドル状態のシステムのオーバーヘッドが解消されます。また、Solaris の以前の印刷機能にあった、ネットワークに接続された印刷サポート用の余分なシステム構成が不要になります。

印刷ファイルにフィルタを適用する

印刷フィルタは、待ち行列内のファイルの内容をある形式から別の形式に変換するプリンタサーバー上のプログラムです。

印刷フィルタは、必要に応じて簡単なものでも複雑なものでもかまいません。SunOS リリースの場合は、出力先プリンタが PostScript 形式へのデータ変換を必要とするほとんどの PostScript 印刷状況に対応する印刷フィルタが、`/usr/lib/lp/postscript` ディレクトリに組み込まれています。PostScript 以外のプリンタ用のフィルタが必要な場合は、そのフィルタを作成し、目的のシステムに追加しなければなりません。

一連の「印刷フィルタ記述子ファイル」が `/etc/lp/fd` ディレクトリに用意されています。これらの記述子ファイルは、フィルタの特性 (高速フィルタや低速フィルタなど) を記述し、フィルタプログラム (`/usr/lib/lp/postscript/postdaisy` など) を指します。

プリンタインタフェースプログラムの機能

LP 印刷サービスは、オペレーティングシステムの他の部分と情報をやり取りします。また、標準プリンタインタフェースプログラムを使用して、次の作業を実行します。

- 必要に応じてプリンタポートを初期化する。標準プリンタインタフェースプログラムは、`stty` コマンドを使用してプリンタポートを初期化する。
- プリンタを初期化する。標準プリンタインタフェースプログラムは、`terminfo` データベースと `TERM` シェル変数を使用して、適切な制御シーケンスを見つける。
- 必要に応じてバナーページを印刷する。
- 印刷要求で指定された部数だけ印刷する。

LP 印刷サービスは、別のプログラムが指定されなければ、標準インタフェースプログラム (`/usr/lib/lp/model` ディレクトリに入っています) を使用します。独自のインタフェースプログラムを作成することもできますが、独自のプログラムがプリンタへの接続を終了させないことや正しいプリンタの初期設定を妨げないことを確認しなければなりません。

lp sched デーモンによる印刷ジョブ状態の確認

`lp sched` デーモンは、処理する各印刷要求のログを保持し、印刷処理中に発生したエラーを記録します。このログは `/var/lp/logs/lpsched` ファイルに保管されます。毎晩、`lp cron` ジョブは `/var/lp/logs/lpsched` ファイルを新しい `lp sched.n` ファイル名に変更し、新しいログファイルを開始します。エラーが発生したり、印刷待ち行列からジョブが消えたりした場合は、ログファイルを使用して `lp sched` デーモンで実行された印刷ジョブへの処理を判別できます。

ログファイルの消去

/var/lp/logs ディレクトリ内の lpsched および requests ログファイルは、情報が追加されるにつれて大きくなります。LP 印刷サービスは、デフォルトの cron ジョブを使用してログファイルを消去します。lp cron ジョブは /var/spool/cron/crontabs/lp ファイルに入っています。このジョブはログファイルの内容を定期的に移動します。ログの内容は log.1 に移動され、log.1 の内容は log.2 に移動されるというようになります。log.2 が上書きされると、その内容は失われます (つまり、前の log.1 の内容が代わりに書き込まれます)。

▼ プリンタ要求のログの交換間隔を変更する方法

Solaris 2.6 リリースより、プリンタサーバー上の requests ログファイルは、毎日ではなく、毎週交換されることになりました。プリンタサーバーの使用頻度が高い場合は、交換間隔を毎日に再設定できます。

1. プリンタサーバー上でスーパーユーザーまたは lp になります。

2. EDITOR 環境変数を設定します。

```
# EDITOR=vi
# export EDITOR
```

3. lp の crontab ファイルを編集します。

```
# crontab -e lp
```

4. requests ログファイルの交換間隔を指定するファイルの先頭行を、毎日曜日 (0) から、毎日を示すアスタリスク (*) に変更します。

```
13 3 * * * cd /var/lp/logs; if [ -f requests ]; then if
[ -f requests.1 ]; then /bin/mv requests.1 requests.2; fi; /usr/bin/cp
requests requests.1;>requests; fi
```

5. ファイルを保存して、終了します。

ローカル印刷の処理スケジュール

印刷要求は、クライアントとサーバーが異なるシステムにある場合と同じ経路をたどります。要求は常に同じ経路をたどり、クライアントからサーバーに流れます。

次の図に、ユーザーがローカルプリンタ上に PostScript ファイルを印刷する要求を出したときに実行される処理を示します。ローカルプリンタとは、ユーザーのシステムに接続されたプリンタです。ローカルシステム上の lpsched デーモンがすべての処理を実行します。これらの処理には、プリンタと内容形式の照合、デフォルトプリンタの特定、要求のフィルタ処理、指定されたプリンタインタフェースプログラムの起動などが含まれます。

プリンタインタフェースプログラムは、次のことを行います。

- パナーページの印刷
- プリンタ障害の検出
- プリンタ障害ポリシーを、リセット、継続、または印刷ジョブのやり直しのどれかに決定する
- 最後に、インタフェースプログラムは `lpcat` プログラムを使用して、印刷要求をローカルプリンタのシリアルポートにダウンロードする

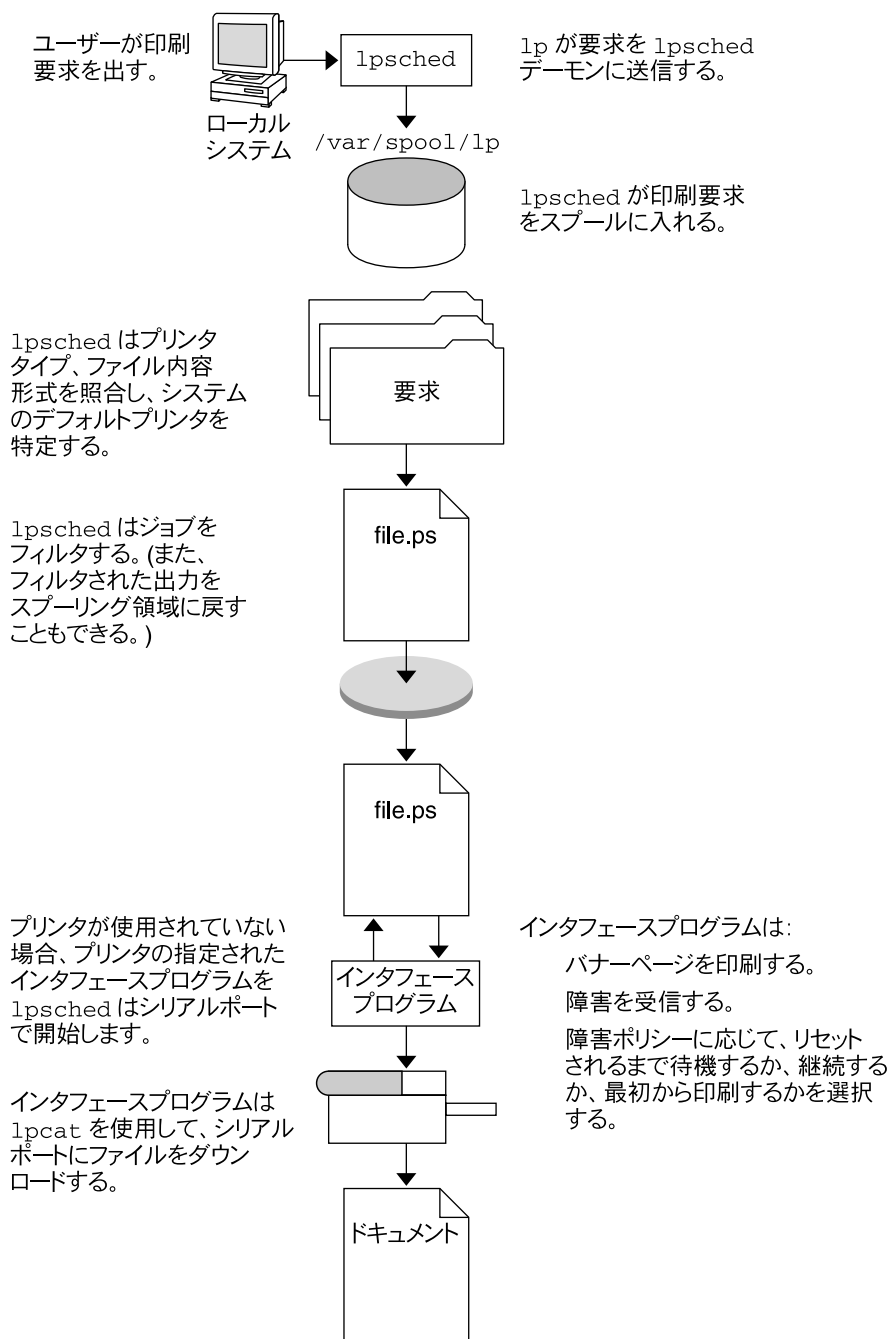


図 8-2 ローカルでの印刷処理

リモート印刷の処理スケジュール

次の図は、Solaris 印刷クライアントのユーザーが lpd ベースのプリンタサーバーに印刷要求を出したときに実行される処理を示します。このコマンドは、直接プリンタサーバーと接続して、自身の通信を処理します。

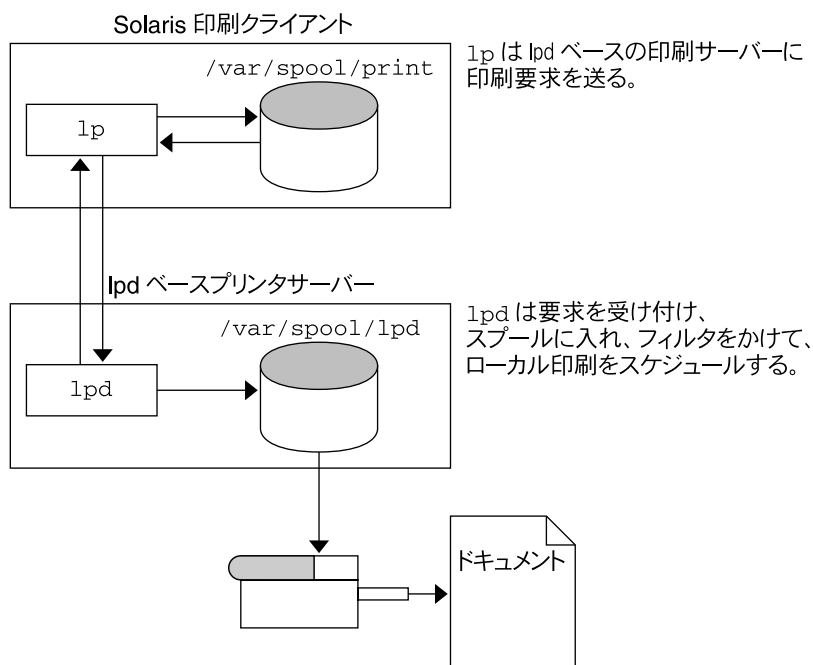


図 8-3 Solaris 印刷クライアントと lpd ベースのプリンタサーバー間の印刷

次の図は、lpd ベースの印刷クライアントが Solaris プリンタサーバーに印刷要求を出す様子を示しています。lpd デーモンは、印刷要求のローカル部分とプリンタサーバーへの接続を処理します。プリンタサーバー上の inetd プロセスは、ネットワーク印刷要求を待って、プロトコルアダプタを起動して要求を処理します。プロトコルアダプタは lpsched デーモンと通信し、このデーモンがプリンタサーバー上で要求を処理します。

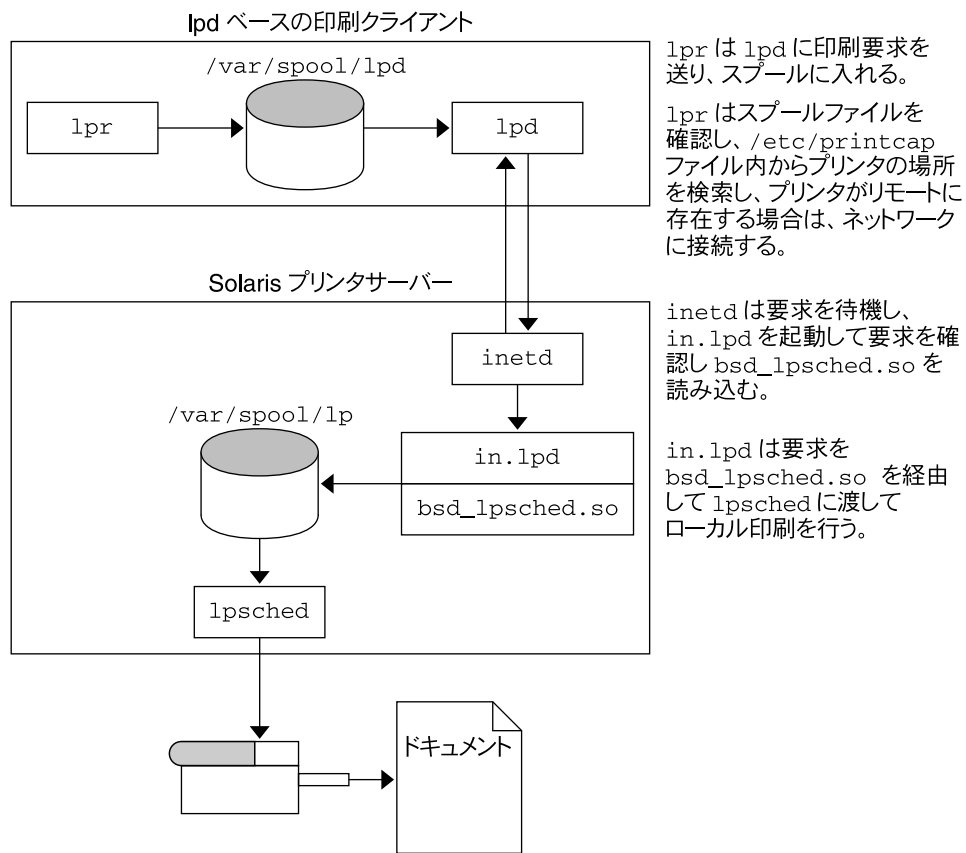


図 8-4 lpd ベースの印刷クライアントと Solaris プリンタサーバー間の印刷

次の図は、Solaris 印刷クライアントのユーザーが Solaris プリンタサーバーに印刷要求を出したときに実行される処理を示します。印刷クライアント上の印刷コマンドは、プリンタサーバーと直接通信することにより、各印刷要求のローカル部分を処理します。

プリンタサーバー上の `inetd` プロセスは、ネットワーク上の印刷要求を監視し、プロトコルアダプタを起動して、プリンタサーバー上の `lpsched` デーモンと通信します。このデーモンが印刷要求を処理します。

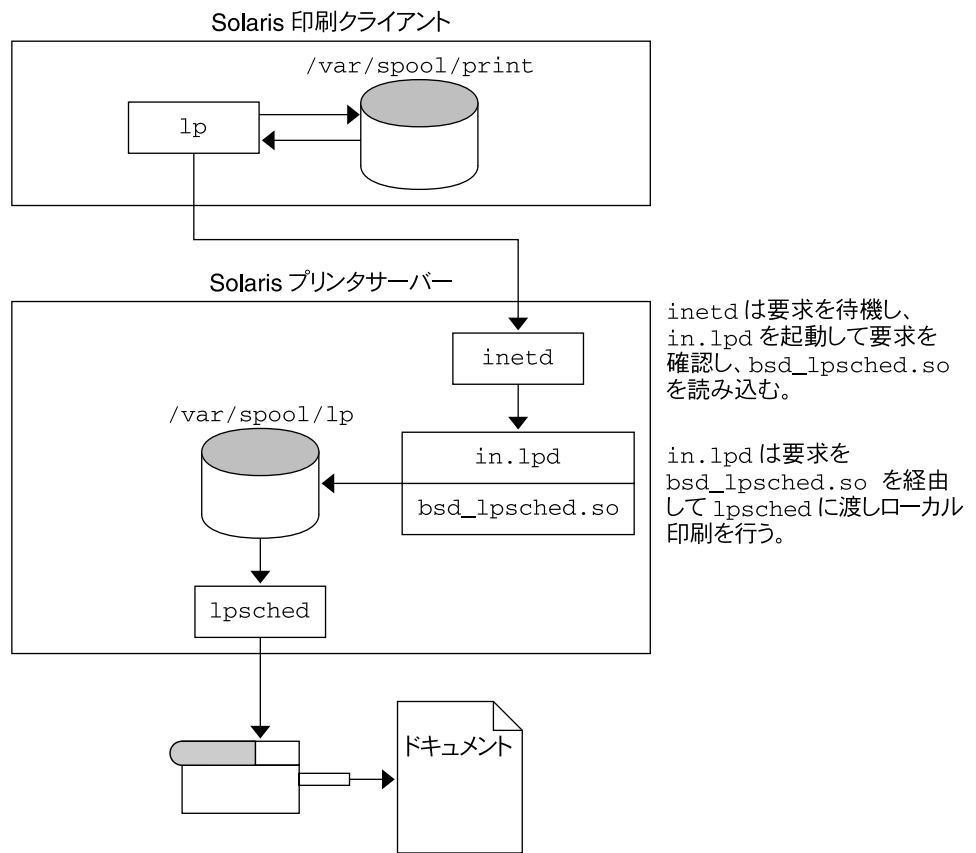


図 8-5 Solaris 印刷クライアントと Solaris プリンタサーバー間の印刷

第 9 章

端末とモデムの管理 (トピック)

以下の各章で、端末とモデムの管理について説明します。

第 10 章	端末とモデムの概要について説明します。
第 11 章	端末とモデムを設定する手順について説明します。
第 12 章	SAF コマンドを使用して端末とモデムを設定する手順について説明します。

第 10 章

端末とモデムの管理 (概要)

この章では、端末やモデムを管理する場合の概要を説明します。この章の内容は次のとおりです。

- 181 ページの「端末、モデム、ポート、サービス」
- 184 ページの「端末とモデムを管理するツール」
- 184 ページの「シリアルポートツール」
- 184 ページの「サービスアクセス機能 (SAF)」

シリアルポートツールで端末とモデムを設定する手順については、第 11 章を参照してください。

SAF で端末とモデムを設定する手順については、第 12 章を参照してください。

端末とモデムの管理に関する新機能

Solaris 管理コンソールは、端末とモデムを設定するためのシリアルポートツールを提供します。Solaris 管理コンソールの起動については、『Solaris のシステム管理 (基本編)』の「Solaris Management Console を起動する」を参照してください。端末やモデムを設定する手順については、Solaris 管理コンソールのオンラインヘルプを参照してください。

端末、モデム、ポート、サービス

端末とモデムは、システム資源とネットワーク資源へのローカルおよびリモートのアクセスを提供します。端末とモデムの設定は、システム管理者の重要な作業です。この節では、Solaris 環境におけるモデムと端末の管理についての概要を説明します。

端末

システムのビットマップグラフィックスディスプレイは、シリアルポートに接続され、テキストしか表示できない英数字端末とは異なります。グラフィックスディスプレイは、特別な手順に従って管理する必要はありません。

モデム

モデムには次の3つの基本構成があります。

- 発信専用
- 着信専用
- 発着信両用

家庭用コンピュータに接続されるモデムの中には、「発信専用」サービス向けに設定されていることがあります。その場合、ユーザーは家から他のコンピュータにアクセスできますが、外からは誰もユーザーのコンピュータにアクセスできません。

「着信専用」サービスは発信専用のちょうど逆です。つまり、リモートサイトからはシステムにアクセスできますが、そのシステムから外側には呼び出しができません。

「発着信両用」アクセスは、その名前が示すとおり、着信専用、発信専用の両機能を持っています。

ポート

「ポート」とは、装置がオペレーティングシステムと通信するためのチャンネルのことです。具体的には、端末やモデムのケーブルを物理的に接続する「コンセント」と考えると一番わかりやすいでしょう。

ただし、ポートは厳密には物理的なコンセントではなく、その実体はハードウェア（ピンとコネクタ）とソフトウェア（デバイスドライバ）からなっています。多くの場合、1つの物理的なコンセントが複数のポートを備えており、複数の装置を接続できます。

一般的なポートとして、シリアル、パラレル、SCSI (Small Computer Systems Interface)、Ethernet などがあります。

「シリアルポート」は、標準の通信プロトコルを使用し、1本の信号線で1バイト単位の情報を1ビットずつ送信します。

RS-232-C または RS-423 規格に準拠して設計されている装置（大部分のモデム、英数字端末、プロッタ、一部のプリンタ）は、同様に設計されたコンピュータのシリアルポートにはどれにでも、標準ケーブルを使用して接続できます。

1 台のコンピュータに多数のシリアルポートを接続する場合は、システムに「アダプタボード」を追加する必要があります。アダプタボードは、ドライバソフトウェアを使用することにより、より多くの装置を接続できるための追加のシリアルポートを提供します。

サービス

モデムや端末を使用すると、シリアルポートのソフトウェアを介してコンピュータ資源にアクセスできます。シリアルポートソフトウェアは、ポートに接続する装置向けに特定の「サービス」を提供するように設定しなければなりません。たとえば、モデムに対してはシリアルポートは発着信両用サービスを提供するように構成できます。

ポートモニター

特定のサービスへのアクセスは、主に「ポートモニター」を通じて行います。ポートモニターとは、ログイン要求や、プリンタまたはファイルのアクセス要求を常に監視しているプログラムのことです。

ポートモニターは要求を検出すると、オペレーティングシステムとサービスを要求する装置間の通信を確立するのに必要なすべてのパラメータを設定します。次に、必要なサービスを提供する他のプロセスに制御を移します。

表 10-1 に、Solaris 環境で提供されている 2 つのタイプのポートモニターとその説明を示します。

表 10-1 ポートモニターのタイプ

ポートモニター	マニュアルページ	説明
listen	listen(1M)	Solaris 2.6 より前のシステムからのリモート印刷要求の処理など、ネットワークサービスへのアクセスを制御する。デフォルトの Solaris オペレーティング環境では、このタイプのポートモニターは使用されていない。
ttymon	ttymon(1M)	モデムや英数字端末が必要とするログインサービスへのアクセスを提供する。シリアルポートツールは、それらの装置からのログイン要求を処理するように、ttymon ポートモニターを自動的に設定する。

getty(1M) という従来のポートモニターに慣れているユーザーも、新しい ttymon を使用してください。新しい ttymon はさらに強力なツールとなっています。1 つの ttymon で複数の getty に相当する処理が行えます。それ以外の点では、どちらのプログラムも同じ機能を提供します。

端末とモデムを管理するツール

次の表は、端末とモデムを管理するツールを列挙したものです。

表 10-2 端末とモデムを管理するツール

ツールの説明	ツール	参照箇所
管理作業全般	サービスアクセス機能 (SAF) のコマンド	184 ページの「サービスアクセス機能 (SAF)」
簡易設定	Solaris 管理コンソールのシリアルポートツール	第 11 章と Solaris 管理コンソールオンラインヘルプ
簡易設定	Admintool	Admintool オンラインヘルプ

シリアルポートツール

シリアルポートツールは、`pmadm` コマンドを呼び出すことにより、シリアルポートソフトウェアを設定して端末やモデムを管理します。シリアルポートツールはまた、次の機能も提供します。

- 共通の端末およびモデム構成用テンプレート
- 複数ポートの設定、変更、または削除
- 各ポートの状態の簡易表示

サービスアクセス機能 (SAF)

SAF は、端末、モデム、その他のネットワーク装置の管理用のツールです。SAF では特に次の設定を行います。

- `ttymon` および `listen` ポートモニター (`sacadm` コマンドを使用)
- `ttymon` ポートモニターサービス (`pmadm`、`ttyadm` コマンドを使用)
- `listen` ポートモニターサービス (`pmadm`、`nlsadmin` コマンドを使用)
- `tty` 装置に関する問題の解決
- ネットワークからの印刷サービス要求に関する問題の解決
- サービスアクセスコントローラに関する問題の解決 (`sacadm` コマンドを使用)

SAF は、`tty` 装置やローカルエリアネットワーク (LAN) を通して行われるシステム資源やネットワーク資源へのアクセスを制御するオープンシステムソリューションです。SAF はプログラムではなく、バックグラウンドプロセスと管理用コマンドの階層構造になっています。

第 11 章

端末とモデムの設定 (手順)

この章では、Solaris 管理コンソールのシリアルポートツールを使用して、端末とモデムを設定する手順を説明します。

この章で説明する手順は次のとおりです。

- 188 ページの「端末を設定する方法」
- 189 ページの「モデムを設定する方法」
- 190 ページの「ポートを初期化する方法」

端末とモデムの概要については、第 10 章を参照してください。

シリアルポートツールによる端末とモデムの設定

Solaris 管理コンソールのシリアルポートツールを使用すると、シリアルポートを設定できます。「シリアルポート (Serial Ports)」メニューからシリアルポートを選択し、「アクション (Action)」メニューから「構成 (Configure)」オプションを選択して、次の内容を設定します。

- 端末
- モデム - 着信
- モデム - 発信
- モデム - 着信 / 発信
- 初期化のみ - 接続なし

「構成 (Configure)」オプションは、上記のサービスを設定するテンプレートへのアクセスを提供します。各シリアルポートの詳細を基本と拡張機能の 2 つのレベルで表示することができます。シリアルポートを選択して、「アクション (Action)」メニューから「プロパティ (Properties)」オプションを選択し、シリアルポートを構成した後、

各シリアルポートの詳細レベルにアクセスすることができます。シリアルポートを構成したら、SAF コマンドでポートを使用可能または使用不可にすることができます。SAF コマンドの使用については、第 12 章を参照してください。

シリアルポートのコマンド行インタフェースについては、`smserialport(1M)` のマニュアルページを参照してください。

端末の設定

表 11-1 にシリアルポートを使用して端末を設定する際のメニュー項目 (およびそれらのデフォルト値) を示します。

表 11-1 端末のデフォルト値

認定レベル	項目	デフォルト値
基本 (Basic)	ポート名 (Port Name)	-
	備考欄 (Description)	Terminal
	サービスの状態 (Service Status)	使用可能
	ボーレート (Baud Rate)	9600
	端末の種類 (Terminal Type)	tvi925
	ログインプロンプト (Login Prompt)	ttyn login:
拡張機能 (Advanced)	キャリア検出 (Carrier Detection)	ソフトウェア
	選択: キャリア検出時に接続 (Connect on Carrier)	いいえ
	選択: 双方向 (Bidirectional)	はい
	選択: 初期化のみ (Initialize Only)	いいえ
	タイムアウト (秒) (Timeout)	なし
	ポートモニター (Port Monitor)	zsmon
	サービスプログラム (Service Program)	/usr/bin/login

モデムの設定

表 11-2 に、シリアルポートでモデムを設定する際に使用できる 3 つのモデム用テンプレートを示します。

表 11-2 モデム用テンプレート

モデム構成	説明
着信専用	モデムに着信はできるが、発信はできない。
発信専用	モデムから発信はできるが、着信はできない。
発着信両用	モデムへ着信も、モデムから発信もできる。

表 11-3 に各テンプレートのデフォルト値を示します。

表 11-3 モデム用テンプレートのデフォルト値

認定レベル	項目	モデム - 着信専用	モデム - 発信専用	モデム - 着信と発信
基本 (Basic)	ポート名 (Port Name)	-	-	-
	備考欄 (Description)	Modem – Dial In Only	Modem – Dial Out Only	Modem – Dial In and Out
	サービスの状態 (Service Status)	使用可能	使用可能	使用可能
	ボーレート (Baud Rate)	9600	9600	9600
	ログインプロンプト (Login Prompt)	ttyn login:	ttyn login:	ttyn login:
拡張機能 (Advanced)	キャリア検出 (Carrier Detection)	ソフトウェア	ソフトウェア	ソフトウェア
	選択: キャリア検出時に接続 (Connect on Carrier)	いいえ	いいえ	いいえ
	選択: 双方向 (Bidirectional)	いいえ	いいえ	はい
	選択: 初期化のみ (Initialize Only)	いいえ	はい	いいえ
	タイムアウト (秒) (Timeout)	なし	なし	なし
	ポートモニター (Port Monitor)	zsmon	zsmon	zsmon
	サービスプログラム (Service Program)	/usr/bin/login	/usr/bin/login	/usr/bin/login

表 11-4 では、「初期化のみ」テンプレートの各デフォルト値を示します。

表 11-4 「初期化のみ (Initialize Only)」のデフォルト値

認定レベル	項目	デフォルト値
基本 (Basic)	ポート名 (Port Name)	-
	説明	Initialize Only - No Connection
	サービスの状態 (Service Status)	有効
	ボーレート (Baud Rate)	9600
	ログインプロンプト (Login Prompt)	ttyn login:
拡張機能 (Advanced)	キャリア検出 (Carrier Detection)	ソフトウェア
	選択: キャリア検出時に接続 (Connect on Carrier)	いいえ
	選択: 双方向 (Bidirectional)	はい
	選択: 初期化のみ (Initialize Only)	はい
	タイムアウト (秒) (Timeout)	なし
	ポートモニター (Port Monitor)	zsmom
	サービスプログラム (Service Program)	/usr/bin/login

▼ 端末を設定する方法

1. Solaris 管理コンソールが実行していなければ、起動します。

```
% /usr/sadm/bin/smc &
```

Solaris 管理コンソールの起動については、『Solaris のシステム管理 (基本編)』の「Solaris Management Console を起動する」を参照してください。

2. ナビゲーション区画で「このコンピュータ (Computer)」アイコンをクリックします。
3. 「Devices and Hardware」→「シリアルポート (Serial Ports)」をクリックします。シリアルポートメニューが表示されます。
4. 端末に使用するポートを選択します。
5. 「アクション (Action)」メニューから「構成 (Configure)」→「端末 (Terminal)」を選択します。
「シリアルポート n を構成 (Configure Serial Port)」ウィンドウが「基本 (Basic)」モードで表示されます。
「端末 (Terminal)」メニュー項目については、表 11-1 を参照してください。

6. 「了解 (OK)」をクリックします。
7. 各項目を設定するには、端末として設定されたポートを選択します。次に、「アクション (Action)」メニューから「プロパティ (Properties)」を選択します。
8. 必要な場合は、テンプレートエントリの値を変更します。
9. 「了解 (OK)」をクリックしてポートを設定します。
10. 端末サービスが追加されていることを確認します。

```
$ pmadm -l -s tty#
```

▼ モデムを設定する方法

1. Solaris 管理コンソールが起動していなければ、起動します。

```
% /usr/sadm/bin/smc &
```

Solaris 管理コンソールの起動については、『Solaris のシステム管理 (基本編)』の「Solaris Management Console を起動する」を参照してください。

2. ナビゲーション区画で「コンピュータ (Computer)」アイコンをクリックします。
3. 「Devices and Hardware」→「シリアルポート (Serial Ports)」をクリックします。シリアルポートメニューが表示されます。
4. モデムに使用するポートを選択します。
5. 「アクション (Action)」メニューから、次の「構成 (Configure)」オプションのいずれかを選択します。
 - a. 「構成 (Configure) (モデム (着信))」
 - b. 「構成 (Configure) (モデム (発信))」
 - c. 「構成 (Configure) (モデム (着信/発信))」

「シリアルポート n を構成 (Configure Serial Port)」ウィンドウが「基本 (Basic)」モードで表示されます。

「モデム (Modem)」メニュー項目については、表 11-3 を参照してください。
6. 「了解 (OK)」をクリックします。
7. 各項目を設定するには、モデムとして設定されたポートを選択します。次に、「アクション (Action)」メニューから「プロパティ (Properties)」を選択します。
8. 必要な場合は、テンプレートエントリの値を変更します。
9. 「了解 (OK)」をクリックしてポートを設定します。
10. モデムのサービスが設定されたことを確認します。

```
$ pmadm -l -s tty#
```

▼ ポートを初期化する方法

1. Solaris 管理コンソールが起動していなければ、起動します。

```
% /usr/sadm/bin/smc &
```

Solaris 管理コンソールの起動については、『Solaris のシステム管理 (基本編)』の「Solaris Management Console を起動する」を参照してください。

2. ナビゲーション区画で「コンピュータ (Computer)」アイコンをクリックします。
3. 「Devices and Hardware」→「シリアルポート (Serial Ports)」をクリックします。シリアルポートメニューが表示されます。
4. 初期化するポートを選択します。
5. 「構成 (Configure)」→「初期化のみ - 接続なし (Initialize Only - No Connection)」を選択します。
「シリアルポート n を構成 (Configure Serial Port)」ウィンドウが「基本 (Basic)」モードで表示されます。
「初期化のみ (Initialize Only)」メニュー項目については、表 11-4 を参照してください。
6. 「了解 (OK)」をクリックします。
7. 各項目を設定するには、初期化のみで設定されたポートを選択します。次に、「アクション (Action)」メニューから「プロパティ (Properties)」を選択します。
8. 必要の場合は、テンプレートエントリの値を変更します。
9. 「了解 (OK)」をクリックしてポートを設定します。
10. モデムのサービスが初期化されたことを確認します。

```
$ pmadm -l -s tty#
```

端末とモデムの問題を解決する方法

端末またはモデムを追加し、適切なサービスを設定したにもかかわらず、シリアルポート回線を通してログインできない場合は、次のような順序で問題を解決してください。

- ユーザーを確認します。

端末やモデムが正しく動作しないという報告は、多くの場合、ログインや着信ができなかったユーザーから寄せられます。したがって、まず、デスクトップに問題がないかどうかを確認することから始めてください。

ログインできない主な原因は、次のとおりです。

- ログイン ID またはパスワードが正しくない
 - 端末が X-ON フロー制御キー (Control-q) の入力を待っている
 - シリアルケーブルの接続が緩んでいるか外れている
 - 端末の設定が正しくない
 - 端末の電源が切られたか、端末に電源が入っていない
- 端末の設定を確認します。

次に、端末またはモデムの設定を調べます。端末またはモデムとの通信の正しい tty 名を調べ、それぞれの設定が tty 名の設定と一致することを確認します。

- 端末サーバーの設定を確認します。

端末に問題のないことがわかったら、端末またはモデムのサーバーに問題がないかどうかを調べます。pmadm コマンドを使用して、ポートモニターが端末またはモデムをサービスするように設定されていて、関連する tty 名が正しいことを確認します。

```
$ pmadm -l -t ttymon
```

/etc/ttydefs を調べ、ラベル定義を端末設定と照合してチェックします。sacadm を使用してポートモニターの状態を調べます。pmadm を使用して、端末が使用するポートのサービスを調べます。

- シリアル接続を確認します。

サービスアクセスコントローラが TTY ポートモニターを起動し、pmadm が端末のポートに対するサービスが有効になっていると報告し、さらに端末の設定がポートモニターの設定と一致する場合は、シリアル接続を調べて問題の原因を探します。シリアル接続は、シリアルポート、ケーブル、端末から構成されています。これらの構成部分のうち 2 つを、信頼性が確認されている他のものに取り替えて、1 箇所ずつテストしてください。

次の構成部分をすべてテストします。

- シリアルポート
 - モデム
 - ケーブル
 - コネクタ
- シリアルポートをコンソールとして使用している場合は、シリアルポートツールからシリアルポートの設定を変更しないでください。コンソール設定を正しく変更するには、/etc/inittab ファイルの次の行を変更してください。

```
co:234:respawn:/usr/lib/saf/ttymon -g -h -p "`uname -n` console
login: " -T terminal_type -d /dev/console -l console -m
ldterm,ttcompat
```


第 12 章

サービスアクセス機能によるシリアルポートの管理 (手順)

この章では、サービスアクセス機能 (SAF) によるシリアルポートサービスの管理方法を説明します。

この章で説明する手順は次のとおりです。

- 201 ページの「ttymon ポートモニターを追加する方法」
- 202 ページの「ttymon ポートモニターの状態を表示する方法」
- 203 ページの「ttymon ポートモニターを停止する方法」
- 203 ページの「ttymon ポートモニターを起動する方法」
- 203 ページの「ttymon ポートモニターを無効にする方法」
- 203 ページの「ttymon ポートモニターを有効にする方法」
- 204 ページの「ttymon ポートモニターを削除する方法」
- 204 ページの「サービスを追加する方法」
- 205 ページの「TTY ポートサービスの状態を表示する方法」
- 208 ページの「ポートモニターサービスを有効にする方法」
- 208 ページの「ポートモニターサービスを無効にする方法」

この章の内容は以下のとおりです。

- 194 ページの「サービスアクセス機能 (SAF) の概要」
- 195 ページの「SAF 全体の管理 (sacadm)」
- 196 ページの「ポートモニターサービス管理 (pmadm)」
- 198 ページの「TTY モニターとネットワークリスナーポートモニター」

SAF については、208 ページの「サービスアクセス機能の管理 (リファレンス)」を参照してください。

サービスアクセス機能 (SAF) の概要

端末とモデムの設定は、Solaris 管理コンソールのシリアルポートツール、`admintool`、`SAF` コマンドで行うことができます。

`SAF` は、端末、モデム、その他のネットワーク装置を管理するためのツールです。`SAF` プログラムの最上位には、サービスアクセスコントローラ (SAC) があります。SAC は、`sacadm` コマンドを使用して管理するポートモニターを制御します。各ポートモニターは1つ以上のポートを管理できます。

管理者は `pmadm` コマンドを使用して、ポートに対応するサービスを管理します。SAC が提供するサービスはネットワークによって異なりますが、SAC と管理コマンド `sacadm` と `pmadm` はネットワークには依存しません。

表 12-1 に `SAF` の制御階層を示します。 `sacadm` コマンドを使用すると、`ttymon` および `listen` ポートモニターを制御する SAC を管理できます。

また、`ttymon` と `listen` のサービスは `pmadm` コマンドにより制御されます。`ttymon` の 1 つのインスタンスは複数のポートにサービスを提供できます。また、`listen` の 1 つのインスタンスは、ネットワークインタフェース上で複数のサービスを提供できます。

表 12-1 `SAF` の制御階層

機能	プログラム	説明
全体の管理	<code>sacadm</code>	ポートモニターの追加および削除用コマンド
サービスアクセスコントローラ	<code>sac</code>	<code>SAF</code> のマスタープログラム
ポートモニター	<code>ttymon</code> <code>listen</code>	シリアルポートのログイン要求を監視する ネットワークのサービス要求を監視する
ポートモニターサービスの管理	<code>pmadm</code>	ポートモニターのサービス制御用コマンド
サービス	ログイン、リモートプロシージャコール、その他	<code>SAF</code> がアクセスを可能にするサービス
コンソールの管理	コンソールログイン	コンソールは、 <code>/etc/inittab</code> ファイル中のエントリにより、 <code>ttymon</code> の直接モードを使用して、自動的に設定される。 <code>pmadm</code> か <code>sacadm</code> を使用して、コンソールを直接管理しないこと。詳細は、199 ページの「 <code>ttymon</code> とコンソールポート」を参照

SAF 全体の管理 (sacadm)

sacadm コマンドは SAF 階層の最上位のコマンドです。sacadm コマンドは主に、ttymon および listen などのポートモニターを追加または削除するのに使用します。このコマンドにはそれ以外に、ポートモニターの現在の状態の表示、ポートモニターの構成スクリプトの管理などの機能があります。

サービスアクセスコントローラ (SAC プログラム)

サービスアクセスコントローラ (SAC) プログラムはすべてのポートモニターを管理します。システムはマルチユーザーモードになると自動的に SAC を起動します。

SAC は、起動されるとまず、各システムの構成スクリプトを探して解釈します。構成スクリプトを使用すると、SAC の環境をカスタマイズできます。このスクリプトは、デフォルトでは空の状態です。ここで行われる SAC の環境に対する変更は、SAC のすべての「子プロセス」に継承されます。継承された環境は継承した子プロセスで変更できます。

SAC プログラムは、システムごとの構成スクリプトの解釈が終わると、SAC の管理ファイルを読み取り、指定されたポートモニターを起動します。各ポートモニターについて、SAC はそれ自身のコピーを実行します (技術的には、SAC が子プロセスをフォークします)。次に、各子プロセスは、それぞれのポートモニターごとの構成スクリプトがあればそれを解釈します。

各ポートモニターの構成スクリプトに指定されている環境を変更すると、それぞれのポートモニターが影響を受け、さらにそれがポートモニターのすべての子プロセスに継承されます。最後に、子プロセスは SAC 管理ファイル内のコマンドを使用して親であるポートモニタープログラムを実行します。

SAC の初期化プロセス

次に、SAC を最初に起動したときに行われる一連の処理を要約します。

1. init が実行レベル 2 で SAC プログラムを生成します。
2. SAC プログラムがシステムごとの構成スクリプト /etc/saf/_sysconfig を読み取ります。
3. SAC プログラムが SAC 管理ファイル /etc/saf/_sactab を読み取ります。
4. SAC プログラムが起動する各ポートモニターの子プロセスをフォークします。
5. 各ポートモニターがポートモニターごとの構成スクリプト /etc/saf/pmtag/_config を読み取ります。

ポートモニターサービス管理 (pmdm)

pmdm コマンドでポートモニターのサービスを管理できます。pmdm コマンドは特にサービスを追加または削除したり、サービスを有効または無効にしたりする場合に使用します。このコマンドでは、さらに、各サービスの構成スクリプトをインストールしたり置き換えたり、サービスに関する情報を出力したりすることもできます。

サービスの各インスタンスは、ポートモニター別、ポート別に一意に識別できなければなりません。pmdm コマンドを使用してサービスを管理する場合、*pmtag* 引数で特定のポートモニターを、また *svctag* 引数で特定のポートをそれぞれ指定します。

ポートモニターのタイプごとに、SAF はポートモニター固有の構成データのフォーマットを定義するための特別なコマンドを必要とします。この構成データは pmdm コマンドで使用します。ttymon および listen ポートモニター用の特別なコマンドは、それぞれ ttyadm と nlsadmin です。

ttymon ポートモニター

直結モデムまたは英数字端末を通してログインしようとするたびに、ttymon は動作を開始します。

図 12-1 に示すように、init プロセスがブート時に最初に起動されるプロセスです。init プロセスは、その管理ファイル (/etc/inittab) を参照して、必要に応じて他のプロセスを起動します。それらのプロセスの 1 つに SAC があります。

SAC が起動されると、今度は SAC がその管理ファイル (/etc/saf/_sactab) に指定されているポートモニターを自動的に起動します。次の図は、1 つの ttymon ポートモニターだけを示しています。

ttymon ポートモニターが起動されると、シリアルポート回線を監視してサービス要求がないかどうかを調べます。

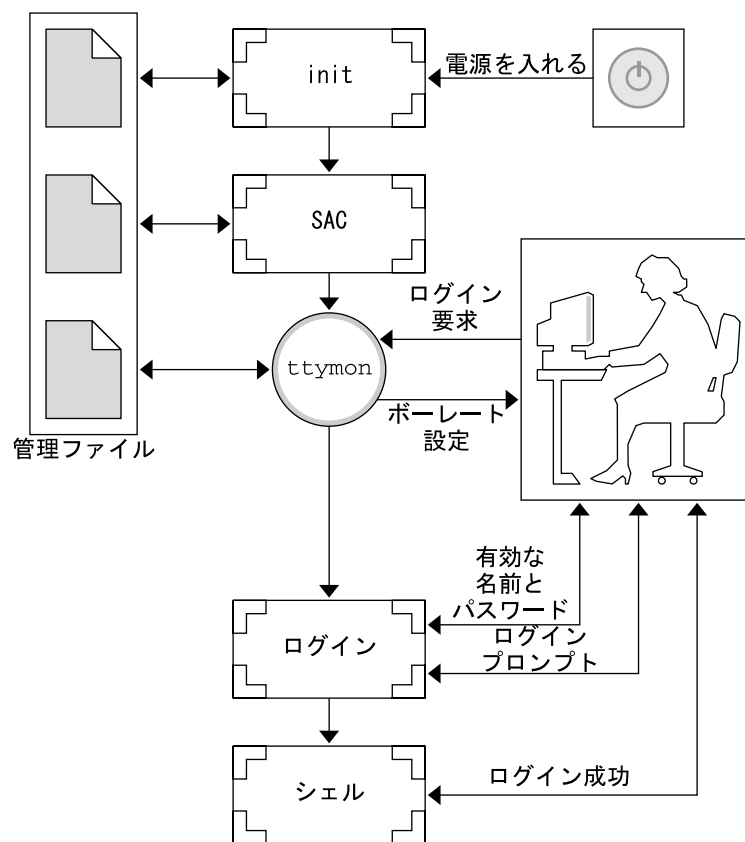


図 12-1 ttymon によるログイン要求の処理

ユーザーが英数字端末やモデムを通してログインしようとする時、シリアルポートドライバはその操作をオペレーティングシステムに伝えます。ttymon ポートモニターはシリアルポートの操作を監視し、通信リンクを確立しようとします。つまり、装置との通信に必要なデータ転送速度、回線制御手順、ハンドシェイクプロトコルを決定します。

モデムや端末との通信用の正しいパラメータの設定が終わると、ttymon ポートモニターはそれらのパラメータをログインプログラムに渡し、制御を移します。

ポートの初期化プロセス

ttymon ポートモニターのインスタンスが SAC によって実行されると、ttymon はポートの監視を始めます。ttymon ポートモニターは、各ポートについて指定されている場合、まず回線制御手順を初期化し、次に回線速度と端末の設定を初期化します。初期化に使用される値は、`/etc/ttydefs` の該当するエントリから得られます。

ttymon ポートモニターは、次に、プロンプトを表示してユーザーからの入力を待ちます。ユーザーが Break キーを押して回線速度が不適當であるという指示を与えると、ttymon ポートモニターは次の速度を設定して、再びプロンプトを表示します。

「自動ボーレート」がポートで有効な場合は、ttymon ポートモニターはそのポートのボーレートを自動的に決めようとします。ttymon ポートモニターがボーレートを認識してプロンプトを表示する前に、ユーザは Return キーを押す必要があります。

有効な入力を受け取ると、ttymon ポートモニターは次のタスクを実行します。

- ポートのサービスごとの構成ファイルを解釈する
- 必要に応じて /etc/utmpx エントリを作成する
- サービス環境を設定する
- ポートに対応したサービスを起動する

サービスが終了すると、ttymon は /etc/utmpx 中にエントリがあれば削除し、ポートを初期状態に戻します。

発着信両用サービス

ポートが発着信両用サービスに設定されている場合、ttymon ポートサービスは次のように行います。

- ユーザーをサービスに接続可能にする。
- uucico、cu、または ct コマンドが、(空いていれば) ポートを発信専用モードで使用できるようにする。
- 文字を読み取ってからプロンプトを表示する。
- 接続要求があると (connect-on-carrier フラグが設定してある場合)、プロンプトメッセージを送らないでポートの対応サービスを起動する。

TTY モニターとネットワークリスナー ポートモニター

SAF は、将来のモニターや他社製のポートモニターの管理に対応するために総合的な管理方法を提供していますが、Solaris 環境では ttymon と listen の 2 つだけが実装されています。

TTY ポートモニター (ttymon)

ttymon ポートモニターは、STREAMS をベースにしており、以下を実行します。

- ポートの監視

- 端末モード、ボーレート、回線制御手順の設定
- ログインプロセスの起動

ttymon ポートモニターは、以前のバージョンの SunOS 4.1 ソフトウェアのもとで `getty` が提供していたのと同じサービスを Solaris ユーザーに提供します。

ttymon ポートモニターは SAC プログラムで実行され、`sacadm` コマンドを使用して構成します。ttymon の各インスタンスはそれぞれに複数のポートを監視できます。それらのポートはポートモニターの管理ファイル内に指定します。また、この管理ファイルは `pmadm` および `ttymax` コマンドを使用して構成します。

ttymon とコンソールポート

コンソールサービスの管理は、サービスアクセスコントローラや、明示的な ttymon 管理ファイルの実行によるものではありません。 `/etc/inittab` ファイル内のエントリを使用し、ttymon を直接モードで使用して、コンソールポートの管理を行います。直接モードとは、ttymon の特別なモードのことで、ログインサービスが必要なコマンドによって直接呼び出されます。

`/etc/inittab` ファイル内のデフォルトのコンソールエントリは次のようになります。

```
co:234:respawn:/usr/lib/saf/ttymon -g -h -p "`uname -n` console login: "
-T terminal_type -d /dev/console -l console -m ldterm,ttcompat
```

`co:234:respawn:`

`co` は、エントリをコンソールとして指定する。234 は動作の実行レベルを指定する。`respawn` は、コンソールエントリが失敗した場合、あるいは実行レベル 2、3、4 に存在しない場合に、コンソールエントリを再起動する必要があることを示す

`/usr/lib/saf/ttymon -g -h`

`-g` オプションを使用するため、正しいボーレートと正しい端末設定をポート上で設定でき、SAC による事前構成なしに、ログインサービスに接続できる。`-h` オプションは、デフォルトまたは指定した速度に設定する前に、回線速度をゼロに設定することにより、回線をハングアップさせる

`-p "`uname -n` console login:`

コンソールポート用のプロンプト文字列を指定する

`-t terminal_type`

コンソールの端末タイプを指定する

-d /dev/console -l console -m
ldterm,ttcompat

-d オプションは、コンソールデバイスを指定する。-l オプションは、
/etc/ttydefsファイル内の ttylabel を指定する。-m オプションは、プッシュする
STREAMS モジュールを指定する

ttymon 固有の管理コマンド (ttyadm)

ttymon の管理ファイルは、sacadm および pmadm の他に ttyadm コマンドによっても更新できます。ttyadm コマンドは、ttymon 固有の情報を書式化し、それらの情報を標準出力に書き出し、書式化された ttymon 固有のデータを sacadm および pmadm コマンドに提示する手段を提供します。

したがって、ttyadm は ttymon を直接管理するのではなく、一般的な管理用コマンドである sacadm および pmadm を補足するものです。詳細は、ttyadm(1M) のマニュアルページを参照してください。

ネットワークリスナーサービス (listen)

listen ポートモニターは SAC プログラムで動作し、以下を実行します。

- ネットワークを監視してサービス要求がないかを調べる
- サービス要求があれば受け付ける
- それらのサービス要求に応答してサーバーを呼び出す

listen ポートモニターは sacadm コマンドを使用して構成します。listen の各インスタンスはそれぞれに複数のサービスを提供できます。それらのサービスは listen ポートモニターの管理ファイルに指定します。この管理ファイルは pmadm および nlsadmin コマンドを使用して構成します。

ネットワークリスナープロセスは、トランスポート層インタフェース (TLI) 仕様に準拠する任意の接続型トランスポートプロバイダで使用できます。Solaris 環境では、listen ポートモニターは、inetd サービスが提供しない追加ネットワークサービスを提供できます。

listen 固有の管理コマンド (nlsadmin)

listen ポートモニターの管理ファイルは、sacadm および pmadm の他に、nlsadmin コマンドでも更新できます。nlsadmin コマンドは、listen 固有の情報の書式を定義して標準出力に書き込み、書式付きの listen 固有のデータを sacadm および pmadm コマンドに提示する手段を提供します。

したがって、nlsadmin は listen を直接管理するのではなく、一般的な管理用コマンドである sacadm および pmadm を補足するものです。

個別に構成される各ネットワークには、ネットワークリスナープロセスのインスタンスが少なくとも1つは存在します。nlsadmin コマンドは listen ポートモニターの動作状態を制御します。

nlsadmin コマンドは、与えられたネットワークに対して listen ポートモニターを設定し、そのポートモニターの固有の属性を構成し、そのモニターを起動したり、強制終了させたりすることができます。さらに、マシン上にある listen ポートモニターについて報告することもできます。

詳細は、nlsadmin(1M) のマニュアルページを参照してください。

ttymon ポートモニターの管理

sacadm コマンドを使用して ttymon ポートモニターを追加、表示、削除、終了、起動、あるいは有効または無効にすることができます。

注 - 次の手順を行うにはスーパーユーザーになる必要があります。

▼ ttymon ポートモニターを追加する方法

ttymon ポートモニターを追加するには、次のように入力します。

```
# sacadm -a -p mbmon -t ttymon -c /usr/lib/saf/ttymon -v `ttyadm  
-V` -y "TTY Ports a & b"
```

- a ポートモニターフラグを追加する。
- p mbmon をポートモニタータグとして指定する。
- t ポートモニタータイプを ttymon として指定する。
- c ポートモニターを起動するのに使用するコマンド文字列を定義する。
- v ポートモニターのバージョン番号を指定する。
- y ポートモニターのインスタンスを説明するコメントを指定する。

▼ ttymon ポートモニターの状態を表示する方法

ttymon ポートモニターの状態を見るには、次のように入力します。

```
# sacadm -l -p mbmon
```

-l ポートモニターを表示する。
-p mbmon をポートモニタータグとして指定する。

例 - ttymon ポートモニターの状態を表示する

```
# sacadm -l -p mbmon
PMTAG  PMTYPE  FLGS RCNT STATUS  COMMAND
mbmon  ttymon  -    0   STARTING /usr/lib/saf/ttymon #TTY Ports a & b
```

PMTAG	ポートモニター名 mbmon を指定する。
mbmon	
PMTYPE	ポートモニターのタイプ ttymon を指定する
ttymon	
FLGS	次の2つのフラグが設定されているかどうかを示す。
-	d は、新しいポートモニターを有効にしない。 x は、新しいポートモニターを起動しない。この例では、どちらのフラグも設定されていない
RCNT	戻りカウント値を示す。0 の戻りカウントは、ポートモニターが失敗した場合でも再起動しないことを示す。
0	
STATUS	ポートモニターの現在の状態を示す。
STARTING	
COMMAND	ポートモニターを起動するコマンドを指定する。
/usr/lib/saf ...	
#TTY Ports a & b	ポートモニターを説明するコメントを指定する。

▼ ttymon ポートモニターを停止する方法

ttymon ポートモニターを終了するには、次のように入力します。

```
# sacadm -k -p mbmon
```

-k ポートモニターを終了する。

-p mbmon をポートモニタータグとして指定する。

▼ ttymon ポートモニターを起動する方法

終了した ttymon ポートモニターを起動するには、次のように入力します。

```
# sacadm -s -p mbmon
```

-s ポートモニターを起動する。

-p mbmon をポートモニタータグとして指定する。

▼ ttymon ポートモニターを無効にする方法

ポートモニターを無効にすると、以前から存在しているサービスをそのまま有効にするため、新しいサービスが起動できなくなります。

ttymon ポートモニターを無効にするには、次のように入力します。

```
# sacadm -d -p mbmon
```

-d ポートモニターを無効にする。

-p mbmon をポートモニタータグとして指定する。

▼ ttymon ポートモニターを有効にする方法

ttymon ポートモニターを有効にすると、そのモニターが新しい要求にサービスを提供できるようになります。

ttymon ポートモニターを有効にするには、次のように入力します。

```
# sacadm -e -p mbmon
```

- e ポートモニターを有効にする。
- p mbmon をポートモニタータグとして指定する。

▼ ttymon ポートモニターを削除する方法

ttymon ポートモニターを削除するには、次のように入力します。

```
# sacadm -r -p mbmon
```

- r ポートモニターを削除する。
- p mbmon をポートモニタータグとして指定する。

注 - ポートモニターを削除すると、それに関連するすべての構成ファイルが削除されます。ポートモニター構成ファイルは sacadm コマンドでは更新や変更ができません。ポートモニターを再構成するには、古いポートモニターを削除してから新しいポートモニターを追加してください。

ttymon サービスの管理

pmadm コマンドを使用してサービスを追加したり、ポートモニターに関連付けられている 1 つ以上のポートのサービスを表示したり、サービスを有効または無効にしたりできます。

注 - 次の手順を行うにはスーパーユーザーでなければなりません。

▼ サービスを追加する方法

標準の端末サービスを mbmon ポートモニターに追加するには、次のように入力します。

```
# pmadm -a -p mbmon -s a -i root -v `ttyadm -V` -m "`ttyadm -i 'Terminal disabled' -l contty -m ldterm,ttcompat -S y -d /dev/term/a -s /usr/bin/login`"
```

注 – 上記の入力例では、`contty` の後が次の行にまたがっていますが、実際には Return キーを押さずに (改行なしに) 入力します。

- a ポートモニターを追加する。
- p **mbmon** をポートモニタータグとして指定する。
- s **a** をポートモニターサービスタグとして指定する
- i 識別情報を、サービス実行中にポートモニターサービスタグに割り当てられるように指定する。
- v ポートモニターのバージョン番号を指定する。
- m **ttyadm** により書式化された **ttymon** 固有の構成データを指定する。

上記の `pmadm` コマンドには `ttyadm` コマンドが組み込まれています。その組み込みコマンドの中の指定項目の意味は次のとおりです。

- b ポートフラグを発着信両用に指定する。
- i 無効応答メッセージを指定する。
- l /etc/ttydefs ファイルにあるどの TTY 名を使用するか指定する。
- m サービスを起動する前にプッシュする STREAMS モジュールを指定する。
- d TTY ポートに使用する装置へのフルパス名を指定する。
- s 接続要求を受信したとき起動するサービスへのフルパス名を指定する。引数が必要な場合、コマンドと引数を二重引用符 (") で囲む。

▼ TTY ポートサービスの状態を表示する方法

`pmadm` コマンドを次に示すように使用して、特定のポートモニターに設定されている 1 つまたはすべての TTY ポートを表示します。

1 つのポートモニターの 1 つのサービスを表示するには、次のように入力します。

```
# pmadm -l -p mbmon -s a
```

- l サービス情報を表示するフラグ
- p **mbmon** をポートモニタータグとして指定する。
- s **a** をポートモニターサービスタグとして指定する

すべてのポートモニターのすべてのサービスを表示するには、次のように入力します。

```
# pmadm -l
```

-l サービス情報を表示するフラグ

1つのポートモニターのすべてのサービスを表示するには、次のように入力します。

```
# pmadm -l -p mbmon
```

-l サービス情報を表示するフラグ

-p mbmon をポートモニタータグとして指定する。

例 - 1つのポートモニターのすべてのサービスを表示する

```
# pmadm -l -p mbmon
PMTAG  PMTYPE  SVCTAG  FLAGS  ID      <PMSPECIFIC>
mbmon  ttymon  a      -      root   /dev/term/a - - /usr/bin/login - contty
ldterm,ttcompat login:  Terminal disabled - y #
```

mbmon	pmadm -p コマンドを使用して設定された、ポートモニター名 mbmon を指定する
ttymon	ポートモニターのタイプ ttymon を指定する
a	pmadm -s コマンドを使用して設定された、サービスタグ値を示す
-	次のフラグが pmadm -f コマンドを使用して設定されているかどうかを指定する x — サービスを有効にしないことを意味する u — サービス用の utmpx エントリを作成することを意味する。この例では、どちらのフラグも設定されていない
root	起動時にサービスに割り当てられた ID を指定する。この値は、pmadm -i コマンドを使用して設定される
<PMSPECIFIC> Information	
/dev/term/a	ttyadm -d コマンドを使用して設定された、TTY ポートパス名を示す

-	<p>次のフラグが <code>ttynam -c -b -h -I -r</code> コマンドを使用して設定されているかどうかを示す。</p> <p>c — ポート用のキャリアフラグに接続を設定する</p> <p>b — ポートが双方向性である (着信トラフィックと発信トラフィックの両方を許可する) と設定する</p> <p>h — 着呼が受信された直後の自動ハンガアップを抑制する</p> <p>I — ポートを初期化する</p> <p>r — ポートから文字を受信するまで、<code>ttymon</code> が <code>login:</code> メッセージを出力しないよう待機させる</p>
-	<p><code>ttynam -r</code> オプションを使用して設定された値を示す。このオプションは、ポートからデータを受信後に、<code>ttymon</code> がプロンプトを表示するときを決定する。カウントが 0 の場合、<code>ttymon</code> は、任意の文字を受信するまで待機する。カウントが 0 より大きい場合、<code>ttymon</code> は、カウント新規行を受信するまで待機する。この例では、値は設定されていない</p>
<code>/usr/bin/login</code>	<p>接続を受信したときに呼び出されるサービスのフルパス名を指定する。この値は、<code>ttynam -s</code> コマンドを使用して設定される</p>
-	<p><code>ttynam -t</code> コマンドのタイムアウト値を指定する。このオプションは、ポートを開くことに成功して、かつ入力データが <i>timeout</i> 秒内に受信されていない場合に、<code>ttymon</code> がポートを閉じることを指定する。この例では、タイムアウト値は設定されていない</p>
<code>contty</code>	<p><code>/etc/ttydefs</code> ファイル中の TTY 名を指定する。この値は、<code>ttynam -l</code> コマンドを使用して設定される</p>
<code>ldterm, ttcompat</code>	<p>プッシュする STREAMS モジュールを指定する。これらのモジュールは、<code>ttynam -m</code> コマンドを使用して設定される</p>
<code>login: Terminal disabled</code>	<p>ポートが無効であるときに表示される、アクティブでないメッセージを指定する。このメッセージは、<code>ttynam -i</code> コマンドを使用して設定される</p>
<code>tvi925</code>	<p><code>ttynam -T</code> コマンドを使用して設定されている場合、端末タイプを指定する。この例では、端末タイプは、<code>tvi925</code></p>
<code>y</code>	<p><code>ttynam -s</code> コマンドを使用して設定されたソフトウェアキャリア値を指定する。<code>n</code> は、ソフトウェアキャリアをオフにする。<code>y</code> は、ソフトウェアキャリアをオンにする。この例では、ソフトウェアキャリアはオン</p>
<code>#</code>	<p><code>pmadm -y</code> コマンドで指定した任意のコメントを指定する。この例では、コメントは存在しない</p>

▼ ポートモニターサービスを有効にする方法

無効になっているポートモニターのサービスを有効にするには、次のように入力します。

```
# pmadm -e -p mbmon -s a
```

- e 有効にする。
- p mbmon をポートモニタータグとして指定する。
- s a をポートモニターサービスタグとして指定する。

▼ ポートモニターサービスを無効にする方法

ポートモニターのサービスを無効にするには、次のように入力します。

```
# pmadm -d -p mbmon -s a
```

- d 無効にする。
- p mbmon をポートモニタータグとして指定する。
- s a をポートモニターサービスタグとして指定する。

サービスアクセス機能の管理 (リファレンス)

SAF の関連ファイル

SAF は構成ファイルを使用しますが、このファイルは `sacadm` および `pmadm` コマンドを使用して変更できます。構成ファイルを手作業で編集する必要はありません。

ファイル名	説明
<code>/etc/saf/_sysconfig</code>	システムごとの構成スクリプト

ファイル名	説明
/etc/saf/_sactab	SAC の管理ファイル。SACが制御するポートモニターの構成データが入っている
/etc/saf/pmtag	ポートモニター <i>pmtag</i> のホームディレクトリ
/etc/saf/pmtag/_config	存在する場合、ポートモニター <i>pmtag</i> のポートモニターごとの構成スクリプト
/etc/saf/pmtag/_pmtab	ポートモニター <i>pmtag</i> の管理ファイル。 <i>pmtag</i> が提供するサービスのポートモニター固有の構成データが入っている
/etc/saf/pmtag/svctag	サービス <i>svctag</i> のサービスごとの構成スクリプト
/var/saf/log	SAC のログファイル
/var/saf/pmtag	<i>pmtag</i> によって作成されるファイルのディレクトリ。たとえば、ログファイルのディレクトリなど

/etc/saf/_sactab ファイル

/etc/saf/_sactabは次のようになります。

```
# VERSION=1
zsmon:ttymon::0:/usr/lib/saf/ttymon      #

# VERSION=1                               サービスアクセス機能のバージョン番号を示す
zsmon                                       ポートモニター名
ttymon                                     ポートモニターのタイプ
::                                         次の2つのフラグが設定されているかどうかを示す。
                                           d — ポートモニターを有効にしない
                                           x — ポートモニターを起動しない。この例では、
                                           どちらのフラグも設定されていない
0                                           戻りコード値を示す。0の戻りカウントは、ポート
                                           モニターが失敗した場合でも再起動しないことを示す
/usr/lib/saf/ttymon                       ポートモニターのパス名を示す
```

/etc/saf/pmtab/_pmtab ファイル

/etc/saf/pmtab/_pmtab ファイル (/etc/saf/zsmon/_pmtab など) は、次のようになります。

```
# VERSION=1
ttya:u:root:reserved:reserved:reserved:/dev/term/a:I::/usr/bin/login::9600:
ldterm,ttcompat:ttya login\: ::tvi925:y:#
```

# VERSION=1	サービスアクセス機能のバージョン番号を示す
ttya	サービスタグを示す
x,u	次のフラグが設定されているかどうかを指定する。 x — サービスを有効にしないことを意味する u — サービス用の utmpx エントリを作成することを意味する。
root	サービスタグに割り当てられた ID を示す
reserved	このフィールドは予約されている
reserved	このフィールドは予約されている
reserved	このフィールドは予約されている
/dev/term/a	TTY ポートパス名を示す
/usr/bin/login	接続を受信したときに呼び出されるサービスのフルパス名を指定する
:c,b,h,I,r:	次のフラグが設定されているかどうかを示す。 c — ポート用のキャリアフラグに接続を設定する b — ポートが双方向性である (着信トラフィックと発信トラフィックの両方を許可する) と設定する h — 着呼が受信された直後の自動ハンガアップを抑制する I — ポートを初期化する r — ポートから文字を受信するまで、ttymon が login: メッセージを出力しないよう待機させる
9600	/etc/ttydefs ファイルに定義されている TTY 名を指定する
ldterm,ttcompat	プッシュする STREAMS モジュールを指定する。
ttya login\:	表示するプロンプトを指定する
:y/n:	

<i>message</i>	任意のアクティブでない (無効な) 応答メッセージを指定する
<i>tvi925</i>	端末タイプを指定する
<i>y</i>	ソフトウェアキャリアが設定されているかどうかを示す (y/n)

サービスの状態

`sacadm` コマンドはサービスの状態を制御します。次の表は、サービスの起こりうる状態について説明したものです。

状態	説明
有効	デフォルト状態 - ポートモニターを追加したとき、サービスが有効になる
無効	デフォルト状態 - ポートモニターを削除したとき、サービスは停止する

特定のサービスの状態を確認するには、次のように入力します。

```
# pmadm -l -p portmon_name -s soctag
```

ポートモニターの状態

`sacadm` コマンドは、`ttymon` および `listen` ポートモニターの状態を制御します。次の表は、起こりうるポートモニターの状態について説明したものです。

状態	説明
起動	デフォルト状態 - ポートモニターは追加されると自動的に起動される
有効	デフォルト状態 - ポートは追加されると自動的にサービス要求を受け付け可能になる
停止	デフォルト状態 - ポートモニターは削除されると自動的に停止する
無効	デフォルト状態 - ポートモニターは削除されると自動的に提供中であったサービスを続行し、新しいサービスの追加を拒否する
起動中	中間状態 - ポートモニターの起動が進行中

状態	説明
停止中	中間状態 - ポートモニターは手作業で終了過程に入っているが、まだシャットダウン手続きは完了していない。ポートモニターが停止状態になるまでの途中の状態
非動作中	アクティブではない状態 - ポートモニターが強制終了された状態。前の動作状態のときに監視していたすべてのポートがアクセス不可になる。外部のユーザーからはポートが無効なのか、非動作状態なのか区別できない
障害	アクティブではない状態 - ポートモニターを起動して動作状態を維持できない

特定のポートモニターの状態を確認するには、次のように入力します。

```
# sacadm -l -p portmon_name
```

ポートの状態

ポートは、ポートを制御するポートモニターの状態によって、有効または無効にできます。

状態	説明
シリアル (ttymon) ポートの状態	
有効	ttymon ポートモニターはポートにプロンプトメッセージを送り、ログインサービスを提供する
無効	ttymon が強制終了されているか、無効の場合のすべてのポートのデフォルト状態。このように指定した場合、接続要求を受け取ると、ttymon は「disabled」メッセージを送信する

第 13 章

システム資源の管理 (トピック)

以下の各章で、システム資源の管理について説明します。

第 14 章	ディスク割り当て制限、cron および at コマンド、アカウントティングプログラムによるシステム資源の管理に役立つ Solaris コマンドとユーティリティの概要を説明します。
第 15 章	システム情報を表示したり、変更したりする手順を説明します。
第 16 章	使用されていないファイルや大きなディレクトリを見つけて、ディスク空間を最適化するための手順を説明します。
第 17 章	ディスク割り当て制限を設定し、管理する手順を説明します。
第 18 章	crontab および at コマンドを使用して、繰り返しのまたは 1 度限りのシステムイベントのスケジュールを設定する手順を説明します。
第 19 章	アカウントティングを設定し、管理する手順を説明します。
第 20 章	システムアカウントティングソフトウェアに関するリファレンス情報を提供します。その中には、さまざまなアカウントティング機能の操作方法、さまざまなアカウントティングプログラムの使用方法などが含まれます。

第 14 章

システム資源の管理 (概要)

この章では、Solaris 9 の新しい機能とシステム資源の管理に役立つ機能について概説します。

これらの機能を使用して、一般的なシステム情報を表示し、ディスク容量を監視し、ディスク割り当て制限を設定し、アカウントティングプログラムを使用し、`crontab` と `at` コマンドをスケジュールして、日常的に使用するコマンドを自動的に実行することができます。

この節では、柔軟な方法でシステム資源の割り当て、監視、制御を可能にする Solaris の資源管理機能については言及しません。

Solaris 資源管理機能を使用しないでシステム資源を管理する手順については、217 ページの「システム資源管理の概要 (作業マップ)」を参照してください。

Solaris 資源管理機能を使用したシステム資源の管理については、『Solaris のシステム管理 (資源管理とネットワークサービス)』の「Solaris 9 リソースマネージャの紹介」を参照してください。

システム資源管理の新機能

df、du、ls の新オプション

df、du、ls コマンドに、新しい `-h` オプションが加わり、ディスクの使用状況とファイルやファイルシステムのサイズを 1024 の累乗で表示します。このオプションは、ファイルやディレクトリのサイズが 1024 バイトを上回る場合に、ディスクの空き容量をキロバイト、メガバイト、ギガバイト、テラバイトで示すことによって、df、du、ls `-l` コマンドの出力の解釈をより簡単にします。

詳細については、df(1B)、du(1)、および ls(1) を参照してください。

このガイドの詳細は、228 ページの「ファイルとディスク容量の情報を表示する方法」を参照してください。

システムパラメータの変更

このマニュアルの前バージョンには、ユーザー 1 人あたりのプロセス数と共有メモリのセグメントを増やすといった、システムパラメータの変更について記述されている節がありました。その節は削除されました。調整可能なシステムパラメータの変更については、『Solaris カーネルのチューンアップ・リファレンスマニュアル』を参照してください。

また、ネットワーク時間サーバーの設定については、『Solaris のシステム管理 (資源管理とネットワークサービス)』の「システムの時刻関連サービス」に移動しました。

Solaris 資源管理

Solaris 資源管理を使用すると、アプリケーションがシステム資源を利用する状況を制御できます。

また、拡張アカウンティング機能を設定して、タスクやプロセスごとのリソースの消費状況を監視したり記録することができます。

拡張アカウンティングでは、アカウンティングデータの一般グループをあらわす、新しい可変長の汎用アカウンティングファイル形式を導入しています。これによって、カーネルによってさまざまなアカウンティングファイルに記録される資源の利用率を設定することができます。

Solaris 資源管理機能には、次のものがあります。

- タスク — リソースの使用状況を追跡するための新しいプロセス集合
- プロジェクト — リソースの使用状況を請求するための新しい管理データベース。実行されたタスクに基づいて、資源をプロジェクトに充てることができます。
- acctadm — 拡張アカウンティング機能の様々な属性を構成するための新しいツール。たとえば、システムのアカウンティングによって追跡された資源は、システム全体に基づいて構成することができます。

デフォルトの新しいアカウンティング構成は管理を必要としません。ただし、拡張アカウンティング機能を使用しなくても、/etc/project ファイルを削除しないでください。このファイルが存在しないと、ユーザーはシステムにログインできません。

このリリースで拡張されたアカウンティング機能についての詳細な情報は、次の表を参照してください。

Solaris 資源管理機能	参照箇所
拡張アカウントिंगのオンとオフ	『Solaris のシステム管理 (資源管理とネットワークサービス)』の「拡張アカウントング」および acctadm(1M) のマニュアルページ
プロジェクトデータベースの記述について	『Solaris のシステム管理 (資源管理とネットワークサービス)』の「プロジェクトとタスク」および project(4) のマニュアルページ
拡張されたアカウントング機能のデータの処理について	libexacct(3LIB)、getacct(2)、putacct(2)、および wracct(2) のマニュアルページ

システム資源管理の概要 (作業マップ)

作業	説明	参照先
システム情報の表示と変更	さまざまなコマンドを使用して、一般的なシステム情報、日付と時間、システムのホスト名の変更など、システム情報を表示したり変更することができます。	第 15 章
ディスク使用状況の管理	ディスク容量の使用状況を識別し、古い使用されていないファイルを削除します。	第 16 章
割り当て制限の管理	UFS ファイルシステム割り当て制限を使用して、どのくらいのディスク容量がユーザーによって使用されるのかを管理します。	第 17 章
システムイベントのスケジュール	cron および at ジョブを使用して、古い使用されていないファイルの整理などのスケジュールシステムの日常的な定型作業に役立ちます。	第 18 章
システムアカウントングの管理	システムアカウントングを使用して、ユーザーとアプリケーションが使用するシステム資源を識別します。	第 19 章

第 15 章

システム情報の表示と変更 (手順)

この章では、最も一般的なシステム情報を表示および変更するために必要な手順を示します。この章で説明する手順は次のとおりです。

- 220 ページの「システムが 64 ビット Solaris オペレーティング環境を実行できるか調べる方法」
- 222 ページの「一般的なシステム情報を表示する方法」
- 223 ページの「システムのホスト ID を表示する方法」
- 223 ページの「日付と時刻を表示する方法」
- 224 ページの「システムの日付と時刻を手作業で設定する方法」
- 225 ページの「その日のメッセージを設定する方法」
- 225 ページの「システムのホスト名の変更方法」

システム情報の表示

表 15-1 に、一般的なシステム情報を表示するためのコマンドを示します。

表 15-1 システム情報を表示するためのコマンド

コマンド	マニュアルページ	表示されるシステム情報
psrinfo	psrinfo(1M)	プロセッサタイプ
isainfo	isainfo(1)	動作しているシステムのネイティブアプリケーションによってサポートされるビット数を報告する。ビット数は、トークンとしてスクリプトに渡すことができる

表 15-1 システム情報を表示するためのコマンド (続き)

コマンド	マニュアルページ	表示されるシステム情報
showrev	showrev(1M)	ホスト名、ホスト ID、リリース、カーネルアーキテクチャ、アプリケーションアーキテクチャ、ハードウェアプロバイダ、ドメイン、およびカーネルのバージョン
uname	uname(1)	オペレーティングシステム名、リリース、バージョン、ノード名、ハードウェア名、およびプロセッサタイプ
hostid	hostid(1)	ホスト ID
prtconf	prtconf(1M)	インストールされているメモリー量
date	date(1)	日付と時刻

▼ システムが 64 ビット Solaris オペレーティング環境を実行できるか調べる方法

現在、64 ビット Solaris オペレーティング環境をサポートするプラットフォームは UltraSPARC システムだけです。システムが UltraSPARC システムかどうかを確認するには、次のコマンドを使用します。

```
$ uname -m
sun4u
```

uname -m コマンドの出力が sun4u の場合、そのマシンは UltraSPARC システムです。

Solaris 9 リリースを実行している場合は、psrinfo コマンドを使用することによって、そのマシンが UltraSPARC システムであることを確認できます。

```
# psrinfo -v
Status of processor 0 as of: 05/30/01 13:48:46
  Processor has been on-line since 05/18/01 10:02:40.
  The sparcv9 processor operates at 333 MHz,
    and has a sparcv9 floating point processor.
```

プロセッサタイプが sparcv9 であれば、そのプラットフォームで 64 ビット Solaris オペレーティング環境が実行できます。以前のバージョンの psrinfo コマンドでは、すべてのプラットフォームがプロセッサタイプ sparc として報告されるため、この検査は機能しません。

▼ 64 ビット Solaris 機能が有効になっているか調べる方法

システムで 64 ビット Solaris 機能が有効になっているか調べるには `isainfo` コマンドが使用できます。有効になっていれば、システムは 64 ビットカーネルでブートされています。

例— 64 ビット Solaris 機能が有効になっているか調べる

32 ビットカーネルを実行している UltraSPARC システムの出力は、次のようになります。

```
$ isainfo -v
32-bit sparc applications
```

この出力は、システムが 32 ビットアプリケーションだけをサポートすることを示します。

64 ビットカーネルを実行している UltraSPARC システムの出力は、次のようになります。

```
$ isainfo -v
64-bit sparcv9 applications
32-bit sparc applications
```

この出力は、システムが 32 ビットと 64 ビットのアプリケーションを両方サポートすることを示しています。

動作しているシステムのネイティブアプリケーションによってサポートされるビット数を表示するには、`isainfo -b` コマンドを使用します。

32 ビット Solaris オペレーティング環境が動作する SPARC、IA、UltraSPARC システムは、次のように表示されます。

```
$ isainfo -b
32
```

64 ビット Solaris オペレーティング環境が動作する 64 ビット UltraSPARC システムは、次のように表示されます。

```
$ isainfo -b
64
```

コマンドは 64 だけを返します。64 ビット UltraSPARC システムでは 32 ビットと 64 ビットのアプリケーションが両方動作しますが、64 ビットシステムで実行するには 64 ビットアプリケーションが最適です。

▼ Solaris リリース情報を表示する方法

Solaris リリースバージョンを特定するには、`/etc/release` ファイルの内容を表示します。

```
$ more /etc/release
                                Solaris 8 2/02 s28s_u7wos_08a SPARC
                                Copyright 2002 Sun Microsystems, Inc. All Rights Reserved.
                                Assembled 18 December 2001
```

▼ 一般的なシステム情報を表示する方法

一般的なシステム情報を表示するには、`showrev` コマンドを使用します。

```
$ showrev [-a]
```

`-a` オプションは、提供可能なあらゆるシステム情報を表示します。

例 — 一般的なシステム情報を表示する

次の例は、`showrev` コマンドの出力を示します。

```
$ showrev -a
Hostname: starbug
Hostid: nnnnnnnn
Release: 5.9
Kernel architecture: sun4u
Application architecture: sparc
Hardware provider: Sun_Microsystems
Domain: solar.com
Kernel version: SunOS 5.9 May 2002

OpenWindows version:
X11 Version 6.4.2 11 April 2001

No patches are installed
$
```

▼ 一般的なシステム情報を表示する方法

システム情報を表示するには、`uname` コマンドを使用します。

```
$ uname [-a]
```

`-a` オプションは、オペレーティングシステム名とともに、システムノード名、オペレーティングシステムのリリース、オペレーティングシステムのバージョン、ハードウェア名、プロセッサタイプを表示する

例 — 一般的なシステム情報を表示する

次の例は、`uname` コマンドの出力を示します。

```
$ uname
SunOS
$ uname -a
SunOS starbug 5.9 Generic sun4u sparc SUNW,Ultra-5_10
$
```

▼ システムのホスト ID を表示する方法

ホスト ID を 16 進形式で表示するには、`hostid` コマンドを使用します。

```
$ hostid
```

例 — システムのホスト ID を表示する

次の例は、`hostid` コマンドの出力を示します。

```
$ hostid
80a5d34c
```

▼ システムにインストールされているメモリーを表示する方法

システムにインストールされているメモリー量を表示するには、`prtconf` コマンドを使用します。

```
$ prtconf [| grep Memory]
```

`grep Memory` コマンドは `prtconf` コマンドの出力内容を選別して、メモリー情報だけを表示します。

例 — システムにインストールされているメモリーを表示する

次の例は、`prtconf` コマンドの出力例を示します。

```
# prtconf | grep Memory
Memory size: 128 Megabytes
```

▼ 日付と時刻を表示する方法

システムクロックに従った現在の日付と時刻を表示するには、`date` コマンドを使用します。

```
$ date
```

例 — 日付と時刻を表示する

次の例は、date コマンドの出力例を示します。

```
$ date
Thu May 31 17:44:58 MDT 2001
$
```

システム情報の変更

この節では、一般的なシステム情報を変更するコマンドを説明します。

▼ システムの日付と時刻を手作業で設定する方法

1. スーパーユーザーになります。
2. 次のように新しい日付と時刻を入力します。

```
# date mmddHHMM[[cc]yy]
```

<i>mm</i>	月。2桁を使用
<i>dd</i>	日。2桁を使用
HH	時。2桁で24時間制を使用
MM	分。2桁を使用
<i>cc</i>	世紀。2桁を使用
<i>yy</i>	年。2桁を使用

詳細は、date (1) のマニュアルページを参照してください。

3. オプションを付けずに date コマンドを使用し、システム日付が正しく設定されたことを確認します。

例 — システムの日付と時刻を手作業で設定する

次の例は、date コマンドを使用して手作業でシステムの日付と時刻を設定する方法を示します。


```
# date
Thu Jun 21 13:59:15 MDT 2001
# date 0621141001
Thu Jun 21 14:10:00 MDT 2001
```

▼ その日のメッセージを設定する方法

その日のメッセージファイル `/etc/motd` を編集して、システムの全ユーザーに対して、ログイン時に通知または問い合わせる内容を書き込みます。ただし、この機能を使用するときは、必要なメッセージだけを送るようにします。メッセージファイルは定期的に編集して、不用になったメッセージを削除することをお勧めします。

1. スーパーユーザーになります。
2. エディタを使って `/etc/motd` ファイルを開き、必要なメッセージを追加します。テキストを編集して、ユーザーのログイン時に表示されるメッセージを記述します。スペース、タブ、リターンも入れてください。
3. `/etc/motd` ファイルの内容を表示して、変更内容を確認します。

```
$ cat /etc/motd
Welcome to the UNIX Universe. Have a nice day.
```

例 — その日のメッセージを設定する

Solaris ソフトウェアのインストール時に、デフォルトのその日のメッセージが設定されます。メッセージの内容は次のような SunOS バージョン情報です。

```
$ cat /etc/motd
Sun Microsystems Inc.    SunOS 5.9          Generic  May 2002
```

次の例は、編集後の `/etc/motd` ファイルの内容を示します。このファイルは、ログインする各ユーザーに対してシステムの利用度に関する情報を提供します。

```
$ cat /etc/motd
The system will be down from 7:00 a.m to 2:00 p.m. on
Saturday, July 7, for upgrades and maintenance.
Do not try to access the system during those hours.
Thank you.
```

▼ システムのホスト名の変更方法

システムのホスト名は、複数の異なる場所に指定します。

ネームサービスデータベースを更新して、新しいホスト名を反映させる必要があることを覚えておいてください。

1. スーパーユーザーになります。
2. 次のファイルでシステムのホスト名を変更します。
 - /etc/nodename
 - /etc/hostname.*xy*
 - /etc/inet/hosts
 - /etc/net/ticlts/hosts
 - /etc/net/ticots/hosts
 - /etc/net/ticotsord/hosts
3. (省略可能) ネームサービスを使用している場合は、host ファイルでシステムのホスト名を変更します。
4. システムをリブートして、新しいホスト名を起動します。

```
# init 6
```

第 16 章

ディスク使用の管理 (手順)

この章では、使用していないファイルや大きなディレクトリを見つけることにより、ディスク容量を最適化する方法を示します。この章で説明する手順は次のとおりです。

この章で説明する手順は次のとおりです。

- 228 ページの「ファイルとディスク容量の情報を表示する方法」
- 230 ページの「ファイルサイズを表示する方法」
- 231 ページの「サイズの大きなファイルを見つける方法」
- 232 ページの「指定されたサイズ制限を超えるファイルを見つける方法」
- 233 ページの「ディレクトリ、サブディレクトリ、およびファイルのサイズを表示する方法」
- 234 ページの「ローカル UFS ファイルシステムのユーザーの所有状況を表示する方法」
- 236 ページの「最新ファイルのリストを表示する方法」
- 236 ページの「古いファイルと使用されていないファイルを見つけて削除する方法」
- 237 ページの「一時ディレクトリを一度にクリアする方法」
- 238 ページの「core ファイルを見つけて削除する方法」
- 239 ページの「クラッシュダンプファイルを削除する方法」

ファイルとディスク容量の情報の表示

次の表は、サイズとディスク容量の情報を表示するコマンドをまとめたものです。

コマンド	マニュアルページ	説明
df	df(1M)	空きディスクブロック数とファイル数を表示する。
du	du(1)	各サブディレクトリに割り当てられたディスク容量を表示する。
find -size	find(1)	-size オプションで指定したサイズに基づいて、ディレクトリを再帰的に検索する。
ls -lh	ls(1)	ファイルのサイズを 1024 の累乗でリストする。

▼ ファイルとディスク容量の情報を表示する方法

次のように df コマンドを入力して、ディスク容量の利用状況に関する情報を表示します。

```
$ df [directory] [-h] [-t] [-t]
```

df	オプションを指定しない場合、マウントされている全ファイルシステム、それらの装置名、使用されている 512 バイトのブロック数、ファイル数のリストを表示する
directory	チェックするファイルシステムのディレクトリを指定する
-h	ディスク容量を 1024 の累乗で表示する
-t	マウントされている全ファイルシステムの合計ブロック数と使用されているブロック数を表示する

例 — ファイルサイズおよびディスク容量に関する情報を表示する

次の例では、/usr/dist を除き、すべてのファイルシステムがローカルでマウントされています。/usr/dist はシステム venus からリモートにマウントされています。

```

$ df
/                (/dev/dsk/c0t0d0s0 ): 287530 blocks   92028 files
/usr             (/dev/dsk/c0t0d0s6 ): 1020214 blocks  268550 files
/proc           (/proc                ):      0 blocks    878 files
/dev/fd         (fd                   ):      0 blocks     0 files
/etc/mnttab    (mnttab              ):      0 blocks     0 files
/var/run       (swap                ): 396016 blocks  9375 files
/tmp           (swap                ): 396016 blocks  9375 files
/opt           (/dev/dsk/c0t0d0s5 ): 381552 blocks  96649 files
/export/home   (/dev/dsk/c0t0d0s7 ): 434364 blocks  108220 files
/usr/dist      (venus:/usr/dist     ):14750510 blocks 2130134 files

```

次の例は、ファイルシステム情報が 1024 バイト単位で表示されています。

```

$ df -h
Filesystem      size  used  avail capacity  Mounted on
/dev/dsk/c0t0d0s0 1.9G   58M   1.8G     4%      /
/dev/dsk/c0t0d0s6 2.5G  765M   1.7G     1%     /usr
/proc            0K     0K     0K     0%     /proc
fd               0K     0K     0K     0%     /dev/fd
mnttab           0K     0K     0K     0%     /etc/mnttab
swap            2.1G   24K   2.1G     1%     /var/run
swap            2.1G   0K    2.1G     0%     /tmp
/dev/dsk/c0t0d0s5 1.9G   12M   1.8G     1%     /opt
/dev/dsk/c0t0d0s7 1.9G   10K   1.8G     1%     /export/home

```

注 - /proc と /tmp は、ローカルファイルシステムですが、UFS ファイルシステムではありません。/proc は PROCFS ファイルシステムであり、/var/run と /tmp は TMPFS ファイルシステムであり、/etc/mnttab は MNTFS ファイルシステムです。

次の例は、マウントされているすべてのファイルシステム、装置名、使用されている 512 バイトブロックの合計数、ファイル数を示しています。2 行構成の各エントリの 2 行目は、それぞれのファイルシステムに割り当てられているブロックの合計数とファイルの合計数を示します。

```

$ df -t
/                (/dev/dsk/c0t0d0s0 ): 287530 blocks   92028 files
                  total: 385614 blocks   96832 files
/usr             (/dev/dsk/c0t0d0s6 ): 1020214 blocks  268550 files
                  total: 2381102 blocks  300288 files
/proc           (/proc                ):      0 blocks    879 files
                  total:      0 blocks    924 files
/dev/fd         (fd                   ):      0 blocks     0 files
                  total:      0 blocks     72 files
/etc/mnttab    (mnttab              ):      0 blocks     0 files
                  total:      0 blocks     1 files
/var/run       (swap                ): 396112 blocks  9375 files
                  total: 396112 blocks  9395 files
/tmp           (swap                ): 396112 blocks  9375 files
                  total: 396128 blocks  9395 files
/opt           (/dev/dsk/c0t0d0s5 ): 381552 blocks  96649 files
                  total: 385614 blocks  96832 files

```

```
/export/home      (/dev/dsk/c0t0d0s7 ):  434364 blocks  108220 files
                   total:  434382 blocks  108224 files
/usr/dist         (venus:/usr/dist   ): 14750510 blocks 2130134 files
                   total: 41225162 blocks 2482176 files
```

ファイルサイズの確認

ls コマンドを使用して、ファイルサイズを調べたりソートしたりできます。また、find コマンドを使用して、サイズの制限を超えているファイルを探することができます。詳細は、ls(1) と find(1) のマニュアルページを参照してください。

▼ ファイルサイズを表示する方法

1. 確認したいファイルがあるディレクトリに移動します。
2. 次のように入力して、ファイルのサイズを表示します。

```
$ ls [-lh] [-s]
```

-l	長形式でファイルとディレクトリのリストを表示し、それぞれのサイズをバイト単位で示す(次の例を参照)
-h	ファイルやディレクトリのサイズが 1024 バイトより大きい場合、ファイルとディレクトリのサイズをキロバイト、メガバイト、ギガバイト、テラバイト単位で示す。このオプションは、-o、-n、-@、-g オプションで表示された出力を修正して、ファイルやディレクトリのサイズを新しい形式で表示する。詳細は ls(1) を参照してください。
-s	ファイルとディレクトリのリストを表示し、それぞれのサイズをブロック単位で示す

例 — ファイルサイズを表示する

次の例は、lastlog と messages が /var/adm ディレクトリ内のその他のファイルよりも大きいことを示します。

```
$ cd /var/adm
$ ls -lh
total 148
drwxrwxr-x  5 adm      adm      512 Nov 26 09:39 acct/
-rw-----  1 uucp     bin      0 Nov 26 09:25 aculog
drwxr-xr-x  2 adm      adm      512 Nov 26 09:25 exact/
```

```

-r--r--r-- 1 root other 342K Nov 26 13:56 lastlog
drwxr-xr-x 2 adm adm 512 Nov 26 09:25 log/
-rw-r--r-- 1 root root 20K Nov 26 13:55 messages
drwxr-xr-x 2 adm adm 512 Nov 26 09:25 passwd/
drwxrwxr-x 2 adm sys 512 Nov 26 09:39 sa/
drwxr-xr-x 2 root sys 512 Nov 26 09:49 sm.bin/
-rw-rw-rw- 1 root bin 0 Nov 26 09:25 spellhist
drwxr-xr-x 2 root sys 512 Nov 26 09:25 streams/
-rw-r--r-- 1 root bin 3.3K Nov 26 13:56 utmpx
-rw-r--r-- 1 root root 0 Nov 26 10:17 vold.log
-rw-r--r-- 1 adm adm 19K Nov 26 13:56 wtmpx

```

次の例は、lpsched.1 ファイルが2ブロックを使用していることを示します。

```

$ cd /var/lp/logs
$ ls -s
total 2          0 lpsched          2 lpsched.1

```

▼ サイズの大きなファイルを見つける方法

1. サイズの大きなファイルを検索したいディレクトリに移動します。
2. 次のように入力して、ファイルのサイズをブロック単位に、最も大きいものから降順に表示します。

```
$ ls -s | sort -nr | more
```

```
sort -nr
```

ファイルのリストをブロックサイズの最も大きなものから降順に並べる

例 — サイズの大きなファイルを見つける

次の例では、lastlog と messages が /var/adm ディレクトリ内で最も大きなファイルです。

```

$ cd /var/adm
$ ls -s | sort -nr | more
48 lastlog
30 messages
24 wtmpx
18 pacct
8 utmpx
2 vold.log
2 sulog
2 sm.bin/
2 sa/
2 passwd/
2 pacct1
2 log/

```

```
2 acct/  
0 spellhist  
0 aculog  
total 144
```

▼ 指定されたサイズ制限を超えるファイルを見つける方法

次のように `find` コマンドを使用して、指定したサイズを超えるファイルを見つけてファイル名を表示します。

```
$ find directory -size +nnn
```

<i>directory</i>	ファイルを検索するディレクトリ
-size <i>+nnn</i>	512 バイトブロック数。このサイズを上回るファイルがリストされる。

例 — 指定されたサイズ制限を超えるファイルを見つける

次の例は、作業中のカレントディレクトリ内の 400 ブロックを超えるファイルをどのように見つけるかを示します。

```
$ find . -size +400 -print  
./Howto/howto.doc  
./Howto/howto.doc.backup  
./Howto/howtotest.doc  
./Routine/routineBackupconcepts.doc  
./Routine/routineIntro.doc  
./Routine/routineTroublefsck.doc  
./record  
./Mail/pagination  
./Config/configPrintadmin.doc  
./Config/configPrintsetup.doc  
./Config/configMailappx.doc  
./Config/configMailconcepts.doc  
./snapshot.rs
```

ディレクトリサイズの確認

du コマンドとそのオプションを使用してディレクトリのサイズを表示できます。さらに quot コマンドを使用すれば、ユーザーアカウントによって占められるローカル UFS ファイルシステム上のディスク容量のサイズを知ることができます。これらのコマンドについては、du (1) および quot (1M) のマニュアルページを参照してください。

▼ ディレクトリ、サブディレクトリ、およびファイルのサイズを表示する方法

次のように du コマンドを入力して、1つ以上のディレクトリ、サブディレクトリ、ファイルのサイズを表示します。サイズは 512 バイトブロック単位で表示されます。

```
$ du [-as] [directory ...]
```

du	指定した各ディレクトリとそれらの下の各サブディレクトリのサイズを合わせて表示する
-a	指定したディレクトリ内の各ファイルと各サブディレクトリのサイズと合計ブロック数を表示する
-s	指定したディレクトリ内の合計ブロック数を表示する
-h	各ディレクトリを 1024 バイト単位で表示する
-H	各ディレクトリを 1000 バイト単位で表示する
directory ...	調べたい1つ以上のディレクトリを指定する

例 — ディレクトリ、サブディレクトリ、およびファイルのサイズを表示する

次の例は、2つのディレクトリの合計サイズを示しています。

```
$ du -s /var/adm /var/spool/lp
130    /var/adm
40     /var/spool/lp
```

次の例は、2つのディレクトリとそれらのすべてのサブディレクトリとファイルのサイズ、および各ディレクトリ内の合計ブロック数を表示します。

```
$ du /var/adm /var/spool/lp
2      /var/adm/log
2      /var/adm/passwd
```

```

2      /var/adm/acct/fiscal
2      /var/adm/acct/nite
2      /var/adm/acct/sum
8      /var/adm/acct
2      /var/adm/sa
2      /var/adm/sm.bin
130    /var/adm
4      /var/spool/lp/admins
2      /var/spool/lp/fifos/private
2      /var/spool/lp/fifos/public
6      /var/spool/lp/fifos
2      /var/spool/lp/requests/starbug
4      /var/spool/lp/requests
2      /var/spool/lp/system
2      /var/spool/lp/tmp/starbug
2      /var/spool/lp/tmp/.net/tmp/starbug
4      /var/spool/lp/tmp/.net/tmp
2      /var/spool/lp/tmp/.net/requests/starbug
4      /var/spool/lp/tmp/.net/requests
10     /var/spool/lp/tmp/.net
14     /var/spool/lp/tmp
40     /var/spool/lp

```

次の例は、ディレクトリのサイズを 1024 バイト単位で示しています。

```

du -h /usr/share/audio
796K  /usr/share/audio/samples/au
797K  /usr/share/audio/samples
798K  /usr/share/audio

```

▼ ローカル UFS ファイルシステムのユーザーの所有状況を表示する方法

1. スーパーユーザーになります。
2. 次のように入力して、ユーザー、ディレクトリまたはファイルシステム、**1024** バイト単位のブロック数を表示します。

```
# quot [-a] [filesystem]
```

-a

マウントされている各 UFS ファイルシステムの全ユーザーと 1024 バイト単位のブロック数を表示する

filesystem

UFS ファイルシステムを指定する。ユーザーと使用されているブロック数が表示される

注 `-quot` コマンドは、ローカル UFS ファイルシステムに対してだけ使用できます。

例 — ローカル UFS ファイルシステムのユーザーの所有状況を表示する

次の例では、ルート (/) ファイルシステムのユーザーが表示されています。次に、マウントされた全 UFS ファイルシステムのユーザーが表示されています。

```
# quot /
/dev/rdisk/c0t0d0s0:
43340  root
   3142  rimmer
     47  uucp
     35  lp
     30  adm
     4   bin
     4   daemon
# quot -a
/dev/rdisk/c0t0d0s0 (/):
43340  root
   3150  rimmer
     47  uucp
     35  lp
     30  adm
     4   bin
     4   daemon
/dev/rdisk/c0t0d0s6 (/usr):
460651  root
206632  bin
   791  uucp
   46  lp
     4  daemon
     1  adm
/dev/rdisk/c0t0d0s7 (/export/home):
9      root
```

古いファイルまたは使用されていないファイルの検索と削除

非常に負荷がかかったファイルシステムを整理する作業として、最近使用されていないファイルの特定と削除があります。使用されていないファイルは `ls` または `find` コマンドを使用して見つけることができます。詳細は、`ls(1)` と `find(1)` のマニュアルページを参照してください。

ディスク容量を節約するその他の方法としては、`/var/tmp` または `/var/spool` 内にあるような一時ファイルを空にしたり、`core` ファイルやクラッシュダンプファイルを削除したりするなどが含まれます。クラッシュダンプファイルの詳細は、第 28 章を参照してください。

▼ 最新ファイルのリストを表示する方法

次のように `ls -t` コマンドを使用して、最も新しく作成または変更されたファイルから順番にファイルのリストを表示します。

```
$ ls -t [directory]
```

`-t` 最新タイムスタンプのファイルを最初にしてソートする
`directory` 検索するディレクトリを指定します。

例 — 最新ファイルのリストを表示する

次の例は、`ls -tl` コマンドを使用して、`/var/adm` ディレクトリ内で作成または変更された最新のファイルを検索する方法を示しています。 `sudo` が最も新しく作成または変更されたファイルです。

```
$ ls -tl /var/adm
total 134
-rw----- 1 root    root      315 Sep 24 14:00 sudo
-r--r--r-- 1 root    other    350700 Sep 22 11:04 lastlog
-rw-r--r-- 1 root    bin      4464 Sep 22 11:04 utmpx
-rw-r--r-- 1 adm     adm      20088 Sep 22 11:04 wtmpx
-rw-r--r-- 1 root    other     0 Sep 19 03:10 messages
-rw-r--r-- 1 root    other     0 Sep 12 03:10 messages.0
-rw-r--r-- 1 root    root     11510 Sep 10 16:13 messages.1
-rw-r--r-- 1 root    root     0 Sep 10 16:12 vold.log
drwxr-xr-x 2 root    sys      512 Sep 10 15:33 sm.bin
drwxrwxr-x 5 adm     adm      512 Sep 10 15:19 acct
drwxrwxr-x 2 adm     sys      512 Sep 10 15:19 sa
-rw----- 1 uucp    bin       0 Sep 10 15:17 aculog
-rw-rw-rw- 1 root    bin       0 Sep 10 15:17 spellhist
drwxr-xr-x 2 adm     adm      512 Sep 10 15:17 log
drwxr-xr-x 2 adm     adm      512 Sep 10 15:17 passwd
```

▼ 古いファイルと使用されていないファイルを見つけて削除する方法

1. スーパーユーザーになります。
2. 次のように入力して、指定した日数の間アクセスのないファイルを見つけて、ファイルにそれらのリストを書き込みます。

```
# find directory -type f [-atime +nnn] [-mtime +nnn] -print > filename
```

<i>directory</i>	ファイルを検索するディレクトリを指定する。このディレクトリの下にあるディレクトリも検索する。
<code>-atime + nnn</code>	指定した日数 (nnn) の間アクセスのないファイルを見つける
<code>-mtime + nnn</code>	指定した日数 (nnn) の間変更のないファイルを見つける
<i>filename</i>	使用されないファイルリストを書き込むファイルを指定する

3. 上の手順でリストに書き込んだ使用されていないファイルを削除します。

```
# rm `cat filename`
```

filename は、前の手順で作成した、使用されていないファイルのリストが入っているファイル

例 — 古いファイルと使用されていないファイルを見つけて削除する

次の例は、`/var/adm` ディレクトリとサブディレクトリ中の過去 60 日にわたってアクセスされていないファイルを示しています。`/var/tmp/deadfiles` ファイルには、使用されていないファイルのリストが含まれます。`rm` コマンドは、これらの使用されていないファイルを削除します。

```
# find /var/adm -type f -atime +60 -print > /var/tmp/deadfiles &
# more /var/tmp/deadfiles
/var/adm/aculog
/var/adm/spellhist
/var/adm/wtmpx
/var/adm/sa/sa13
/var/adm/sa/sa27
/var/adm/sa/sa11
/var/adm/sa/sa23
/var/adm/sulog
/var/adm/vold.log
/var/adm/messages.1
/var/adm/messages.2
/var/adm/messages.3
# rm `cat /var/tmp/deadfiles`
#
```

▼ 一時ディレクトリを一度にクリアする方法

1. スーパーユーザーになります。
2. 整理するディレクトリに移動します。

```
# cd directory
```



注意 – 手順 3 を実行する前に、正しいディレクトリにいることを確認してください。
手順 3 はカレントディレクトリ内のすべてのファイルを削除します。

3. カレントディレクトリ内のファイルとサブディレクトリを削除します。

```
# rm -r *
```

4. 不要な一時サブディレクトリやファイル、または古いサブディレクトリやファイルがある他のディレクトリに移動して、手順 3 を繰り返してサブディレクトリやファイルを削除します。

例 — 一時ディレクトリを一度にクリアする

次の例は、どのように mywork ディレクトリを整理するかを示し、すべてのファイルとサブディレクトリが削除されたことを確認する方法を示しています。

```
# cd mywork
# ls
filea.000
fileb.000
filec.001
# rm -r *
# ls
#
```

▼ core ファイルを見つけて削除する方法

1. スーパーユーザーになります。
2. core ファイルを検索したいディレクトリに移動します。
3. 次のように入力して、ディレクトリとサブディレクトリ内のすべての core ファイルを見つけて削除します。

```
# find . -name core -exec rm {} \;
```

例 — core ファイルを見つけて削除する

次の例は、どのように find コマンドを使用して jones のユーザーアカウントから core ファイルを見つけて削除するかを示します。

```
# cd /home/jones
# find . -name core -exec rm {} \;
```

▼ クラッシュダンプファイルを削除する方法

クラッシュダンプファイルは非常に大きくなる可能性があります。したがって、必要以上に長期間保存しないでください。

1. スーパーユーザーになります。
2. 次のように入力して、クラッシュダンプファイルが格納されているディレクトリに移動します。

```
# cd /var/crash/system
```

system は、クラッシュダンプファイルを作成したシステムのことです。



注意 – 手順 3 を実行する前に、正しいディレクトリにいることを確認してください。手順 3 はカレントディレクトリ内のすべてのファイルを削除します。

3. クラッシュダンプファイルを削除します。

```
# rm *
```

4. クラッシュダンプファイルが削除されていることを確認します。

```
# ls
```

例 — クラッシュダンプファイルを削除する

次の例は、システム *venus* からどのようにクラッシュダンプファイルを削除するかと、クラッシュダンプファイルが削除されているかを確認する方法を示します。

```
# cd /var/crash/venus
```

```
# rm *
```

```
# ls
```

```
#
```


第 17 章

ディスク割り当て制限の管理 (手順)

この章では、ディスクの割り当て制限を設定し管理する方法を示します。この章で説明する手順は次のとおりです。

- 245 ページの「割り当て制限用にファイルシステムを構成する方法」
- 246 ページの「1 ユーザーに割り当て制限を設定する方法」
- 247 ページの「複数ユーザーに割り当て制限を設定する方法」
- 248 ページの「割り当て制限の整合性を確認する方法」
- 249 ページの「割り当て制限を有効にする方法」
- 250 ページの「割り当て制限を超過したかどうかを確認する方法」
- 250 ページの「ファイルシステムの割り当て制限を確認する方法」
- 252 ページの「弱い期間制限値のデフォルトを変更する方法」
- 253 ページの「1 ユーザーの割り当て制限を変更する方法」
- 254 ページの「1 ユーザーの割り当て制限を無効にする方法」
- 255 ページの「割り当て制限を無効にする方法」

ディスクの割り当て制限

ディスクの割り当て機能を使用することにより、システム管理者は、各ユーザーが使用できるディスク容量と i ノード数 (おおよそのファイル数に該当) を制限して UFS ファイルシステムを制御できます。このため、ディスク割り当て制限は、特に、ユーザーのホームディレクトリが存在するファイルシステムで便利です。公開ファイルシステムと /tmp ファイルシステムについてはこのディスク割り当て機能による利点はあまりありません。

ディスク割り当て制限の使用

一度割り当て制限を設定しても、それを変更して、ユーザーが使用できるディスク容量と i ノード数を調整できます。また、システムに変更が必要な場合は、それに合わせて割り当て制限を追加または削除できます。ディスク割り当て制限、またはディスク割り当て制限を超えることができる時間を変更する手順、個々のディスク割り当て制限を無効にする手順、あるいはディスク割り当て制限をファイルシステムから削除する手順については、252 ページの「割り当て制限の変更と削除」を参照してください。

ディスク割り当て制限を監視できます。割り当て制限コマンドを使用することによりシステム管理者は、ファイルシステムでの割り当て制限を表示したり、割り当て制限を超えて使用しているユーザーを検索したりできます。これらのコマンドの使用方法については、249 ページの「割り当て制限のチェック」を参照してください。

ディスク割り当て制限の弱い制限値と強い制限値の設定

弱い制限値と強い制限値の両方を設定できます。システムは、ユーザーが自分の強い制限値を超えることを許可しません。しかし、システム管理者は、ユーザーが一時的に超えることができる、弱い制限値を設定できます。弱い制限値は、強い制限値より小さくなければなりません。

いったんユーザーが弱い制限値を超えると、タイマーが起動します。タイマーが動いている間、ユーザーは弱い制限値を超えて操作できます。しかし、強い制限値は超えることができません。再びユーザーが弱い制限値を下回ると、タイマーはリセットされます。しかし、タイマーが期限切れになったときに、まだユーザーの使用率が弱い制限値を超えていた場合、弱い制限値は、強い制限値として実施されます。デフォルトでは、弱い制限値のタイマーは7日です。

`repquota` コマンドと `quota` コマンドの `timeleft` フィールドは、タイマーの値を示します。

たとえば、あるユーザーの弱い制限値が 10,000 ブロックで、強い制限値が 12,000 ブロックであると仮定します。そのユーザーのブロック使用率が 10,000 ブロックを超えて、7日間のタイマーも期限切れになった場合、そのユーザーは自分の使用率が弱い制限値を下回るまで、それ以上のディスクブロックをそのファイルシステム上に割り当てることはできません。

ディスクブロックとファイル制限の相違

ファイルシステムは、ユーザーに2つの資源を提供します。データ用のブロックと、ファイル用のiノードです。各ファイルは、1つのiノードを使用します。ファイルデータは、データブロック内に格納されます(通常は、1Kバイトブロックで構成される)。

ディレクトリがなくても、ユーザーは空のファイルを作成することによって(ブロックを使用することなく)、自分のiノードディスク割り当て制限を超過することができます。また、ユーザーは、ユーザーの割り当て制限のデータブロックをすべて消費するのに十分な大きさのファイルを1つ作成して、自分のブロックのディスク割り当て制限を超過したiノードを1つ使用することができます。

ディスク割り当て制限の設定

割り当て制限を設定する一般的な手順は次のとおりです。

1. 割り当て制限オプションを/etc/vfstabファイルエントリに追加し、quotasファイルをファイルシステムの最上位ディレクトリに作成して、システムをリブートするたびにディスクの割り当て制限が適用されるようにします。
2. まず1人のユーザー用にディスク割り当て制限を作成したら、それをプロトタイプとして他のユーザー用にコピーできます。
3. ディスク割り当て制限を有効にする前に、その割り当て制限と現在のディスク使用状況との整合性をチェックして、矛盾しないことを確認します。
4. 最後に、1つ以上のファイルシステムのディスク割り当て制限を有効にします。

上記手順の具体的な情報については、245ページの「割り当て制限の設定(作業マップ)」を参照してください。

表 17-1 で、ディスク割り当て制限を設定するコマンドを説明します。

表 17-1 割り当て制限を行うコマンド

コマンド	手順	マニュアルページ
edquota	各ユーザーのiノード数とディスク容量に強い制限と弱い制限を設定する	edquota(1M)

表 17-1 割り当て制限を行うコマンド (続き)

コマンド	手順	マニュアルページ
quotacheck	マウントされている各 UFS ファイルシステムを調べ、ファイルシステムのディスク割り当て制限ファイルに格納されている情報とファイルシステムの現在のディスク使用状況を比較し、矛盾を明らかにする	quotacheck (1M)
quotaon	指定したファイルシステムの割り当て制限を有効にする	quotaon (1M)
quota	マウントされているファイルシステムのユーザーの割り当て制限を表示し、割り当て制限が正しく設定されていることを確認する	quota (1M)

割り当て制限設定のガイドライン

ユーザーの割り当て制限を設定する前に、各ユーザーに割り当てるディスク容量の大きさとファイル数を決定する必要があります。ファイルシステムの合計領域サイズを超えないようにする場合は、ファイルシステムの合計サイズをユーザー数に等分すればよいでしょう。たとえば、3人のユーザーが100Mバイトのスライスを共有し、それぞれが同じディスク容量のサイズを必要とする場合は、各ユーザーに33Mバイトずつ割り当てます。

すべてのユーザーがそれぞれに割り当て制限を押し上げることがないような環境では、割り当て制限の合計がファイルシステムの合計サイズを超えるように個々の割り当て制限を設定することも可能です。たとえば、3人のユーザーが100Mバイトのスライスを共有する場合は、各ユーザーに40Mバイトを割り当ててもよいということです。

あるユーザーについて `edquota` コマンドを使用して割り当て制限を決定すると、同じファイルシステム上の他のユーザーにも同じ割り当て制限プロトタイプとして利用できます。

割り当て制限を有効にする前に、まず UFS ファイルシステムの割り当て制限を構成し、各ユーザーの割り当て制限を設定して、`quotacheck` コマンドを実行することによって、現在のディスク使用状況と割り当て制限ファイル間の整合性をチェックします。また、システムのレポートの頻度が低い場合は、`quotacheck` コマンドを定期的に行います。

`edquota` により設定した割り当て制限は、`quotaon` コマンドを使用して有効にしなければ強制的に設定されません。割り当て制限ファイルを正しく構成したら、システムがリブートし、そのファイルシステムがマウントされるたびに、割り当て制限は自動的に有効になります。

割り当て制限の設定 (作業マップ)

作業	説明	参照先
1. ファイルシステムの割り当て制限の構成	/etc/vfstab を編集して、ファイルシステムがマウントされるたびに割り当て制限が有効になるようにする。また、quotas ファイルを作成する	245 ページの「割り当て制限用にファイルシステムを構成する方法」
2. 1 ユーザー用の割り当て制限の設定	edquota を使用して 1 ユーザーアカウント用にディスクと i ノードの割り当て制限を行う	246 ページの「1 ユーザーに割り当て制限を設定する方法」
3. (省略可能) 複数ユーザーの割り当て制限の設定	edquota コマンドを使用して、その他のユーザーアカウント用にプロトタイプの割り当て制限を適用する	247 ページの「複数ユーザーに割り当て制限を設定する方法」
4. 整合性のチェック	quotacheck を使用して、1 つまたは複数のファイルシステムの整合性について、現在の使用状況とディスクの割り当て制限を比較する	248 ページの「割り当て制限の整合性を確認する方法」
5. 割り当て制限を有効にする	quotaon を使用して、1 つまたは複数のファイルシステムの割り当て制限を有効にする	249 ページの「割り当て制限を有効にする方法」

▼ 割り当て制限用にファイルシステムを構成する方法

1. スーパーユーザーになります。
2. /etc/vfstab ファイルを編集します。割り当て制限を設定しようとする各 UFS ファイルシステムの「mount options」フィールドに `rq` を追加します。
3. 割り当て制限を格納しようとするファイルシステムのルートディレクトリに移動します。
4. 次のように入力して、quotas というファイルを作成します。

```
# touch quotas
```
5. **root** にのみ、読み取り権/書き込み権を与えます。

```
# chmod 600 quotas
```

次に進む手順

次の表を見て、次に進む手順を決めてください。

作業	参照箇所
1 ユーザーまたは複数ユーザー用の割り当て制限を設定する	246 ページの「1 ユーザーに割り当て制限を設定する方法」 247 ページの「複数ユーザーに割り当て制限を設定する方法」
既存ファイルを持つファイルシステム上に割り当て制限を設定する場合は、quotacheck コマンドを実行する	248 ページの「割り当て制限の整合性を確認する方法」
ファイルシステムの割り当て制限を有効にする	249 ページの「割り当て制限を有効にする方法」

例 — 割り当て制限用にファイルシステムを構成する

次の `/etc/vfstab` の例は、システム `pluto` の `/export/home` ディレクトリが、ローカルシステムの NFS ファイルシステムとしてマウントされていることを示しています。割り当て制限が使用可能であるかは、`mount options` 列の下の `rq` エントリによってわかります。

```
#device          device  mount      FS   fsck  mount  mount
#to mount        to fsck  point      type pass  at boot options
#
pluto:/export/home -      /export/home nfs   -    yes   rq
```

次の例は `/etc/vfstab` の内容で、割り当て制限が有効 (`mount options` 列の `rq` エントリで示される) なローカル UFS ファイルシステムが `/work` ディレクトリにマウントされていることを示しています。

```
#device          device          mount  FS   fsck  mount  mount
#to mount        to fsck          point  type pass  at boot options
#
/dev/dsk/c0t4d0s0 /dev/rdisk/c0t4d0s0 /work ufs  3    yes   rq
```

▼ 1 ユーザーに割り当て制限を設定する方法

1. スーパーユーザーになります。
2. 割り当て制限エディタを使用して、`quotas` ファイルがファイルシステムのルートディレクトリにある各マウント済み UFS ファイルシステムに対して、1 行の割り当て制限情報を含む一時ファイルを作成します。

```
# edquota username
```

username は、割り当て制限を設定するユーザーです。

3. 1K バイトディスクブロック数の弱い制限値と強い制限値、および i ノード数の弱い制限値と強い制限値を、それぞれ 0 (デフォルト) から各ファイルシステム用に指定されている割り当て値に変更します。

4. ユーザーの割り当て制限を確認します。

```
# quota -v username
```

```
-v                ディスク割り当て制限がある、マウント済みのファイルシステム  
                  上の、ユーザーのディスク割り当て制限情報を表示する
```

```
username         ディスク割り当て制限を表示するユーザー名を指定する
```

例 — 1 ユーザーに割り当て制限を設定する

次の例は、edquota コマンドで開いた一時ファイルの内容を示しています。このシステムでは、ルートディレクトリに quotas ファイルが含まれているマウント済みファイルシステムは、/files だけです。

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

次の例は、割り当て制限設定後の一時ファイルの上と同じ行を示しています。

```
fs /files blocks (soft = 50, hard = 60) inodes (soft = 90, hard = 100)
```

▼ 複数ユーザーに割り当て制限を設定する方法

1. スーパーユーザーになります。
2. 次のように割り当て制限エディタを使用して、すでにプロトタイプユーザー用に設定した割り当て制限を指定するその他のユーザーに適用します。

```
# edquota -p prototype-user username ...
```

```
prototype-user   すでに割り当て制限を設定してあるアカウントのユーザー名
```

```
username ..      1人以上の追加アカウントのユーザー名を指定する
```

例 — 複数ユーザーにプロトタイプ割り当て制限を設定する

次の例は、ユーザー bob に設定された割り当て制限をユーザー mary と john に適用する方法を示しています。

```
# edquota -p bob mary john
```

▼ 割り当て制限の整合性を確認する方法

システムをリブートすると、quotacheck コマンドが自動的に実行されます。通常、割り当て制限を設定したファイルシステムが空の場合、quotacheck コマンドを実行する必要はありません。ただし、既存ファイルを持つファイルシステム上に割り当て制限を設定した場合は、quotacheck コマンドを実行して、既存ファイルや既存 i ノードと割り当て制限データベースとを同期させる必要があります。

また、quotacheck コマンドを大規模ファイルシステムで実行するには、時間がかかることを考慮してください。

注 - ディスクのデータの正確さを保つには、quotacheck コマンドを実行するとき、チェックするファイルシステムが他のユーザーによって使用できないようにしてください。

1. スーパーユーザーになります。
2. 次のように **UFS** ファイルシステム上の整合性チェックを実行します。

```
# quotacheck [-va] filesystem
```

-v	(省略可能) 特定のファイルシステム上の各ユーザーのディスク割り当て制限を示す
-a	/etc/vfstab ファイルに rq エントリがある全ファイルシステムをチェックする
filesystem	チェックするファイルシステムを指定する

詳細は、quotacheck (1M) のマニュアルページを参照してください。

例 — 割り当て制限の整合性を確認する

次の例は、スライス /dev/rdisk/c0t0d0s7 上の /export/home ファイルシステムのディスク割り当て制限をチェックする方法を示しています。/export/home ファイルシステムは、/etc/vfstab ファイルに rq エントリを持つ、唯一のファイルシステムです。

```
# quotacheck -va
*** Checking quotas for /dev/rdisk/c0t0d0s7 (/export/home)
```


▼ 割り当て制限を有効にする方法

1. スーパーユーザーになります。
2. ファイルシステムの割り当て制限を有効にします。

```
# quotaon [-v] -a filesystem ...
```

-v	割り当て制限が無効にされた場合、各ファイルシステムからメッセージを表示する
-a	/etc/vfstab ファイル内に <code>rq</code> エントリがある全ファイルシステムの割り当て制限を有効にする
filesystem ...	指定する 1 つ以上のファイルシステムの割り当て制限を有効にするスペースでファイル名を区切って、複数のファイルシステムを指定する

例 — 割り当て制限を有効にする

次の例は、スライス `/dev/dsk/c0t4d0s7` と `/dev/dsk/c0t3d0s7` 上のファイルシステムのディスク割り当て制限を有効にする方法を示しています。

```
# quotaon -v /dev/dsk/c0t4d0s7 /dev/dsk/c0t3d0s7
/dev/dsk/c0t4d0s7: quotas turned on
/dev/dsk/c0t3d0s7: quotas turned on
```

割り当て制限のチェック

ディスクと i ノードの割り当て制限を設定して有効にしたら、それらの割り当て制限を超過して使用しているユーザーをチェックできます。また、ファイルシステム全体の割り当て制限情報をチェックすることもできます。

表 17-2 で、ディスク割り当て制限をチェックするコマンドを説明します。

表 17-2 割り当て制限チェック用コマンド

コマンド	手順
<code>quota</code>	ユーザー割り当て制限と現在のディスク使用量を表示する。ユーザーの割り当て制限超過使用量も表示可能
<code>repquota</code>	指定されたファイルシステムの割り当て制限、ファイル、および所有しているディスク容量を表示する

1. スーパーユーザーになります。
2. ディスクがまったく使用されていないなくても、1つまたは複数のファイルシステムのすべての割り当て制限を表示します。

```
# repquota [-v] -a filesystem
```

-v	資源を消費していないユーザーも含めて、すべてのユーザーのディスク割り当て制限を報告する
-a	すべてのファイルシステムについて報告する
filesystem	指定したファイルシステムについて報告する

例 — ファイルシステムの割り当て制限を確認する

次の例は、割り当て制限が1つのファイルシステム (/export/home) だけに対して有効なシステムでの repquota コマンドからの出力を示しています。

```
# repquota -va
/dev/dsk/c0t3d0s7 (/export/home):
      Block limits
User      used  soft  hard  timeleft  used  soft  hard  timeleft
#301  --      0     1    2.0 days  0     2     3
#341  --    57    50    60    7.0 days  2     90   100
```

Block limits

used	現在のブロック使用数
soft	弱いブロック制限値
hard	強いブロック制限値
timeleft	ディスク割り当て制限タイマーの残り時間 (日単位)

File limits

used	現在の i ノード使用数
soft	弱い i ノード制限値
hard	強い i ノード制限値
timeleft	ディスク割り当て制限タイマーの残り時間 (日単位)

割り当て制限の変更と削除

割り当て制限を変更して、ユーザーが使用するディスク容量と i ノード数を調整できます。または、必要に応じて各ユーザーから、あるいはファイルシステム全体から割り当て制限を削除できます。

表 17-3 で、割り当て制限を変更または削除するのに使用するコマンドを示します。

表 17-3 割り当て制限を変更または削除するコマンド

コマンド	マニュアルページ	説明
edquota	edquota	各ユーザーについて i ノード数とディスク容量の強い制限と弱い制限を変更する。また、任意のユーザーが弱い制限値を超えることが許される期間の長さを変更する
quotaoff	quotaoff (1M)	指定したファイルシステムの割り当て制限を無効にする

▼ 弱い期間制限値のデフォルトを変更する方法

デフォルトでは、ユーザーはある週の割り当て制限に対する弱い制限値を 1 週間超えることができます。弱い制限値を 1 週間よりも長く超えると、システムはそのユーザーに対し、i ノードとディスクブロックの使用を禁止します。

edquota コマンドを使用すると、この割り当て制限の期間制限を変更できます。

1. スーパーユーザーになります。
2. 次のように割り当て制限エディタを使用して、弱い期間制限値を含む一時ファイルを作成します。

```
# edquota -t
```

-t オプションは、各ファイルシステムの弱い期間制限値を指定します。

3. 期間制限を、0 (デフォルト) から数値とキーワード month、week、day、hour、min、または sec を使用して指定する値に変更します。

注 - この手順は、現在のディスク割り当て制限違反者には影響しません。

例 — 期間の弱い制限値のデフォルトを変更する

次の例は、`/export/home` がただ 1 つの `quotas` を持つマウント済みファイルシステムであるシステムで `edquota` コマンドによって開かれた一時ファイルの内容を示しています。デフォルト値 0 は、デフォルトで 1 週間の期間制限値が使用されることを意味します。

```
fs /export/home blocks time limit = 0 (default), files time limit = 0 (default)
```

次の例は、ブロック割り当て制限の超過に対する期間制限値が 2 週間に変更され、ファイル数の超過に対する期間制限値が 16 日に変更された後の、上の例と同じ一時ファイルの内容を示しています。

```
fs /export/home blocks time limit = 2 weeks, files time limit = 16 days
```

▼ 1 ユーザーの割り当て制限を変更する方法

1. スーパーユーザーになります。
2. 次のように割り当て制限エディタを使用して、`quotas` ファイルがそれぞれのファイルシステムのルートディレクトリにある各マウント済みファイルシステムに対して 1 行ずつエントリが入っている一時ファイルを開きます。

```
# edquota username
```

`username` は、割り当て制限を変更したいユーザー名を指定します。



注意 - `edquota` コマンドの引数として複数のユーザーを指定できますが、表示される情報にはどのユーザーのものなのか示されないため、混乱を招く恐れがあります。

3. **1K** バイトディスクブロック数の弱い制限値と強い制限値、および **i** ノード数の弱い制限値と強い制限値を入力します。
4. ユーザーの割り当て制限が正しく変更されたことを確認します。

```
# quota -v username
```

```
-v
```

ディスク割り当て制限が有効にされている、すべてのマウント済みのファイルシステムについて、ユーザーのディスク割り当て制限情報を表示します

```
username
```

割り当て制限をチェックしたいユーザー名を指定する。

例 — 1 ユーザーの割り当て制限を変更する

次の例は、`edquota` コマンドで開いた一時ファイルの内容を示しています。このシステムでは、ルートディレクトリに `quotas` ファイルが含まれているマウント済みファイルシステムは、`/files` だけです。

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

次の例は、上と同じ一時ファイルの割り当て制限変更後の内容を示しています。

```
fs /files blocks (soft = 0, hard = 500) inodes (soft = 0, hard = 100)
```

次の例は、ユーザー `smith` の強い制限値の変更結果を確認する方法と、1K バイトブロック数と `i` ノード数の強い制限値がそれぞれ 500 と 100 に変更されていることを示しています。

```
# quota -v smith
Disk quotas for smith (uid 12):
Filesystem  usage  quota  limit  timeleft  files  quota  limit  timeleft
 /files      1      0      500      1         0      100
```

▼ 1 ユーザーの割り当て制限を無効にする方法

1. スーパーユーザーになります。
2. 次のように割り当て制限エディタを使用して、`quotas` ファイルがその最上位ディレクトリにある各マウント済みファイルシステムに対して 1 行の割り当て制限情報を含む一時ファイルを作成します。

```
# edquota username
```

`username` は、割り当て制限を無効にしたいユーザー名を指定します。



注意 – `edquota` コマンドの引数として複数のユーザーを指定できますが、表示される情報にはどのユーザーのものなのか示されないため、混乱を招く恐れがあります。

3. 1K バイトディスクブロック数の弱い制限値と強い制限値、および `i` ノード数の弱い制限値と強い制限値を 0 (ゼロ) に変更します。

注 – 必ずこれらの値を 0 (ゼロ) に変更してください。テキストファイルから行を削除してはいけません。

4. ユーザーの割り当て制限を無効にしたことを確認します。

```
# quota -v username
```

<code>-v</code>	ディスク割り当て制限が有効にされている、すべてのマウント済みのファイルシステムについて、ユーザーのディスク割り当て制限情報を表示します
<code>username</code>	割り当て制限を確認したいユーザー名 (UID) を指定します。

例 — 1 ユーザーの割り当て制限を無効にする

次の例は、`edquota` コマンドで開いた一時ファイルの内容を示しています。このシステムでは、ルートディレクトリに `quotas` ファイルが含まれているマウント済みファイルシステムは、`/files` だけです。

```
fs /files blocks (soft = 50, hard = 60) inodes (soft = 90, hard = 100)
```

次の例は、割り当て制限を無効にした後の上と同じ一時ファイルの内容を示しています。

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

▼ 割り当て制限を無効にする方法

1. スーパーユーザーになります。
2. ファイルシステムの割り当て制限を無効にします。

```
# quotaoff [-v] -a filesystem ...
```

<code>-v</code>	割り当て制限が無効にされた場合、各ファイルシステムからメッセージを表示する
<code>-a</code>	全ファイルシステムの割り当て制限を無効にする
<code>filesystem</code>	指定する 1 つ以上のファイルシステムの割り当て制限を無効にするスペースでファイル名を区切って、複数のファイルシステムを指定する

例—割り当て制限を無効にする

次の例は、`/export/home` ファイルシステムの割り当て制限を無効にする方法を示しています。

```
# quotaoff -v /export/home
/export/home: quotas turned off
```


第 18 章

システムタスクのスケジュール設定 (手順)

この章では、`crontab` コマンドおよび `at` コマンドを使用して、定型作業や 1 度限りのシステムタスクをスケジュールリングする方法を説明します。さらに、`cron.deny`、`cron.allow`、`at.deny` の各ファイルを使用して、これらのコマンドの使用を制御する方法も説明します。

この章で説明する手順は次のとおりです。

- 262 ページの「`crontab` ファイルを作成または編集する方法」
- 264 ページの「`crontab` ファイルを表示する方法」
- 265 ページの「`crontab` ファイルを削除する方法」
- 267 ページの「`crontab` の使用を拒否する方法」
- 267 ページの「`crontab` の使用を特定のユーザーに限定する方法」
- 270 ページの「`at` ジョブを作成する方法」
- 271 ページの「`at` 待ち行列を表示する方法」
- 271 ページの「`at` ジョブを表示する方法」
- 272 ページの「`at` ジョブを削除する方法」
- 273 ページの「`at` コマンドの使用を拒否する方法」

システムタスクを自動的に実行する方法

たくさんのシステムタスクを設定して、自動的に実行することができます。これらのタスクの中には、定期的な実行が必要になる作業があります。また、夜間や週末などの就業時間外に 1 回だけ実行する作業もあります。

この節では、`crontab` と `at` という 2 つのコマンドについて概説します。これらのコマンドを使用すると、定型作業をスケジュールして、自動的に実行することができます。`crontab` は、繰り返し実行するコマンドをスケジュールします。`at` コマンドは、1 回だけ実行するタスクをスケジュールします。

次の表は、`crontab` コマンドと `at` コマンド、これらのコマンドの使用を制御できるファイルをまとめたものです。

表 18-1 システムタスクのスケジューリング用コマンド

コマンド	スケジューリングの対象	ファイルの格納場所	制御ファイル
<code>crontab</code>	一定間隔で実行する複数のシステムタスク	<code>/var/spool/cron/crontabs</code>	<code>/etc/cron.d/cron.allow</code> <code>/etc/cron.d/cron.deny</code>
<code>at</code>	1つのシステムタスク	<code>/var/spool/cron/atjobs</code>	<code>/etc/cron.d/at.deny</code>

Solaris 管理コンソールのスケジュールされたジョブツールを使用して、定型作業をスケジュールすることもできます。Solaris 管理コンソールの使用と起動の方法については、『Solaris のシステム管理 (基本編)』の「Solaris Management Console の操作 (手順)」を参照してください。

繰り返されるジョブのスケジューリング (crontab)

定型的なシステム管理タスクは、`crontab` コマンドを使用して、毎日、毎週、または毎月それぞれ 1 回ずつ実行するようにスケジュールできます。

毎日 1 回の `crontab` によるシステム管理作業には、次のようなものがあります。

- 作成後、数日以上経過したファイルを一時ディレクトリから削除する
- アカウンティング要約コマンドを実行する
- `df` および `ps` コマンドを使用してシステムのスナップショットを取る
- 日常のセキュリティ監視を実行する
- システムのバックアップを実行する

毎週 1 回の `crontab` システム管理作業には、次のようなものがあります。

- `man -k` コマンドで処理する `catman` データベースを再構築する
- `fsck -n` を実行して存在するディスク問題のリストを表示する

毎月 1 回の `crontab` システム管理作業には、次のようなものがあります。

- 指定月に使用されなかったファイルをリストする
- 月次アカウンティングレポートを生成する

上記に加えて、連絡事項の通知の転送やバックアップファイルの削除や、さらに他の定型的システム作業を実行するように `crontab` コマンドをスケジュールすることもできます。

crontab ジョブをスケジューリングする手順については、262 ページの「crontab ファイルを作成または編集する方法」を参照してください。

1 つのジョブのスケジューリング (at)

特定の 1 つのジョブを後で実行するように at コマンドを使用してスケジュールできます。ジョブは 1 つのコマンドやスクリプトで構成されます。

crontab コマンドと同様に、at コマンドは定型作業の自動実行をスケジュールします。しかし、crontab ファイルとは異なり、at ファイルはそれぞれのタスクを 1 回実行して、その後はディレクトリから削除されてしまいます。したがって、at はそれぞれ単純なコマンドまたはスクリプトを実行して、後で調べることができるようにそれらの出力を別々のファイルに送るように使用するのが最も効果的です。

at ジョブの実行を指定するには、単にコマンド構文に従って at オプションで実行時刻を指定してください。at ジョブの実行を指定する方法についての詳細は、269 ページの「at コマンドの説明」を参照してください。

at コマンドは、入力されたコマンドまたはスクリプトを、現在の環境変数のコピーと一緒に /var/spool/cron/atjobs ディレクトリに格納します。作成された at ジョブには、ファイル名として、at 待ち行列内での位置を指定する長い数値と .a 拡張子からなる、たとえば 793962000.a のような文字列が与えられます。

cron デーモンは、通常 15 分間隔で定期的に atrun プログラムを実行します。atrun プログラムは、スケジュールされた時間に at ジョブを実行します。cron デーモンが at ジョブを実行すると、at ジョブのファイルが atjobs ディレクトリから削除されます。

at ジョブをスケジューリングする手順については、270 ページの「at ジョブを作成する方法」を参照してください。

繰り返されるシステムタスクのスケジューリング (cron)

以降の各項で、crontab ファイルをどのように作成、編集、表示、削除するか、さらに、それらのファイルの使用をどのように制御するかを説明します。

crontab ファイルの内容

cron デーモンは、各 crontab ファイル内にあるコマンドに従ってシステムタスクをスケジュールします。crontab ファイルには、それぞれ一定間隔で実行されるコマンドが 1 行に 1 つずつ入っています。各行の先頭は cron デーモンが各コマンドを実行する日時情報です。

たとえば、SunOS ソフトウェアのインストール時に root という名前の crontab ファイルが提供されますが、このファイルの内容は次のとおりです。

```
10 3 * * * /usr/sbin/logadm
15 3 * * 0 /usr/lib/fs/nfs/nfsfind
1 2 * * * [ -x /usr/sbin/rtc ] && /usr/sbin/rtc -c > /dev/null 2>&1
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean
```

最初の行は、毎日午前 3 時 10 分に logadm コマンドを実行します。2 行目は、毎週日曜日の午前 3 時 15 分に nfsfind スクリプトを実行します。3 行目は、毎日午前 2 時 10 分に夏時間をチェック (して必要に応じて修正) するスクリプトを実行します。RTC タイムゾーンも /etc/rtc_config ファイルもない場合、このエントリは何もしません。4 行目は、毎日午前 3 時 30 分に Generic Security Service テーブル /etc/gss/gsscred_db の重複エントリをチェック (重複エントリがある場合は削除) します。

crontab ファイル内のコマンド行の構文の詳細は、261 ページの「crontab ファイルエントリの構文」を参照してください。

crontab ファイルは、/var/spool/cron/crontabs ディレクトリに保存されます。SunOS ソフトウェアのインストール時には、root 以外にもいくつかの crontab ファイルが提供されます (表 18-2 を参照してください)。

表 18-2 デフォルトの crontab ファイル

crontab ファイル	機能
adm	アカウントティング
lp	印刷
root	一般的なシステム機能とファイルシステムの整理
sys	パフォーマンス情報の収集
uucp	一般的な uucp の整理

デフォルトの crontab ファイルの他に、ユーザーは crontab ファイルを作成してユーザー自身のシステムタスクをスケジュールできます。その他の crontab ファイルは、それらの中に作成されるユーザーのアカウントに基づいて、bob、mary、smith、jones などのように命名されます。

root または他のユーザーが所有する crontab ファイルに使用するには、スーパーユーザーの特権が必要です。

crontab ファイルの作成、編集、表示、削除の手順については、以降の節で説明します。

cron デーモンのスケジューリング

cron デーモンは crontab コマンドの自動スケジューリングを管理します。cron デーモンは、通常、15 分ごとに /var/spool/cron/crontab ディレクトリに crontab ファイルがあるかどうかをチェックします。cron デーモンは、新しい crontab ファイルがないか、既存の crontab が変更されていないかを確認し、いずれかがあった場合は、ファイル内のリストから実行時刻を読み取り、正しい時刻にコマンドを実行します。

ほとんど同様に、cron デーモンは /var/spool/cron/atjobs ディレクトリ内の at ファイルのスケジューリングを制御します。

crontab ファイルエントリの構文

crontab ファイルは、1 行に 1 つのコマンドが入った構成になっています。これらのコマンド行の最初の 5 つのフィールドには、コマンドが実行される時刻を指定し、それぞれスペースで区切ります。これら 5 つのフィールドを、次の表で説明します。

表 18-3 crontab 時刻フィールドの値

時刻フィールド	値
分	0-59
時	0-23
日	1-31
月	1-12
曜日	0 - 6 (0 は日曜日)

次に、crontab 時刻フィールドで特殊文字を使用する際のガイドラインを示します。

- 各フィールドはスペースで区切る
- 複数の値の間はコンマで区切る
- 値の範囲はハイフンを使用して指定する
- 取り得るすべての値を含むには、ワイルドカードとしてアスタリスクを使用する
- コメントまたは空白行を示すには、行の先頭にコメント記号 (#) を使用する

たとえば、次の crontab コマンドエントリの例は、毎月 1 日と 15 日の午後 4 時に、ユーザーのコンソールウィンドウに注意を促すメッセージを表示します。

```
0 16 1,15 * * echo Timesheets Due> /dev/console
```

crontab ファイル内の各コマンドは、長くても 1 行内に入れなければなりません。これは、crontab が余分なキャリッジリターンを認識しないからです。crontab のエントリとコマンドオプションの詳細は、crontab(1) のマニュアルページを参照してください。

crontab ファイルの作成と編集

crontab ファイルを作成するもっとも簡単な方法は、crontab -e コマンドを使用することです。このコマンドは、使用するシステム環境で使用できる環境変数 EDITOR で指定されたテキストエディタを起動します。この環境変数が設定されていない場合は、crontab はデフォルトのエディタ ed を使用します。あらかじめ、使い慣れたエディタを選択しておく必要があります。

次の例は、エディタが定義されたかどうかを確認する方法と、vi をデフォルトのエディタとして設定する方法を示しています。

```
$ which $EDITOR
$
$ EDITOR=vi
$ export EDITOR
```

crontab ファイルを作成すると、自動的に /var/spool/cron/crontabs ディレクトリ内に格納され、作成者のユーザー名で命名されます。スーパーユーザー特権があれば、他のユーザーや root の crontab ファイルを作成または編集できます。

▼ crontab ファイルを作成または編集する方法

1. (省略可能) root または他のユーザーが所有する crontab ファイルを作成または編集する場合は、スーパーユーザーになります。
2. 次のように入力して、新しい crontab ファイルを作成するか、既存の crontab ファイルを編集します。

```
$ crontab -e [username]
```

username は、crontab ファイルを作成または編集するユーザーのアカウント名を指定します。他のユーザーの crontab ファイルを作成または編集するには、スーパーユーザーの権限が必要です。



注意 – 誤ってオプションを指定しないで `crontab` コマンドを入力した場合は、使用しているエディタの中断文字を入力してください。この文字を入力すると、変更結果を保存せずに `crontab` コマンドを終了できます。この場合に変更結果を保存してファイルを終了すると、既存の `crontab` ファイルが空のファイルで上書きされます。

3. コマンド行をファイルに追加します。

261 ページの「`crontab` ファイルエントリの構文」に記載されている構文に従ってください。`crontab` ファイルは `/var/spool/cron/crontabs` に格納されます。

4. `crontab` ファイルの変更箇所を確認します。

```
# crontab -l [username]
```

例 — `crontab` ファイルを作成または編集する

次の例は、他のユーザーのための `crontab` ファイルをどのように作成するかを示します。

```
# crontab -e jones
```

次の新しい `crontab` ファイルに追加されたコマンドは、毎週日曜日の午前 1 時にユーザーのホームディレクトリからすべてのログファイルを自動的に削除します。このコマンドエントリは出力先を変更しないので、出力先変更文字がコマンド行の `*.log` の後に追加されて、そのコマンドが正しく実行されるようにしています。

```
# This command helps clean up user accounts.
1 0 * * 0 rm /home/jones/*.log> /dev/null 2>&1
```

▼ `crontab` ファイルを確認する方法

特定のユーザーの `crontab` ファイルがあるかどうかを確認するには、`/var/spool/cron/crontabs` ディレクトリで `ls -l` コマンドを使用します。たとえば、次の表示はユーザー `smith` と `jones` の `crontab` ファイルがあることを示しています。

```
$ ls -l /var/spool/cron/crontabs
-rw-r--r-- 1 root sys 190 Feb 26 16:23 adm
-rw----- 1 root staff 225 Mar 1 9:19 jones
-rw-r--r-- 1 root root 1063 Feb 26 16:23 lp
-rw-r--r-- 1 root sys 441 Feb 26 16:25 root
-rw----- 1 root staff 60 Mar 1 9:15 smith
-rw-r--r-- 1 root sys 308 Feb 26 16:23 sys
```

ユーザーの `crontab` ファイルの内容を確認するには、`crontab -l` コマンドを使用します。264 ページの「`crontab` ファイルを表示する方法」を参照してください。

crontab ファイルの表示

crontab -l コマンドは、cat コマンドが他のファイルタイプの内容を表示するのとまったく同様に、使用しているユーザーの crontab ファイルの内容を表示します。このコマンドを使用するために、ディレクトリを (crontab ファイルが入っている) /var/spool/cron/crontabs に変更する必要はありません。

デフォルトでは、crontab -l コマンドは自分自身の crontab ファイルを表示します。他のユーザーの crontab ファイルは、スーパーユーザーでなければ表示できません。

▼ crontab ファイルを表示する方法

1. (省略可能) root または他のユーザーの crontab ファイルを表示する場合は、スーパーユーザーになります。
2. 次のように入力して、crontab ファイルを表示します。

```
$ crontab -l [username]
```

username は、表示する crontab ファイルのユーザーのアカウント名を指定します。他のユーザーの crontab ファイルを表示するには、スーパーユーザーの権限が必要です。



注意 – 誤ってオプションを指定しないで crontab コマンドを入力した場合は、使用しているエディタの中断文字を入力してください。この文字を入力すると、変更結果を保存せずに crontab コマンドを終了できます。この場合に変更結果を保存してファイルを終了すると、既存の crontab ファイルが空のファイルで上書きされます。

例 — crontab ファイルを表示する

次の例で、どのように crontab -l コマンドを使用してユーザのデフォルトの crontab ファイル、root のデフォルトの crontab ファイル、他のユーザーの crontab ファイルを表示するかを示します。

```
$ crontab -l
13 13 * * * chmod g+w /home1/documents/*.book> /dev/null 2>&1
$ suPassword:
# crontab -l
#ident "@(#)root 1.19 98/07/06 SMI" /* SVr4.0 1.1.3.1 */
#
# The root crontab should be used to perform accounting data collection.
```



```
#
# The rtc command is run to adjust the real time clock if and when
# daylight savings time changes.
#
10 3 * * * /usr/sbin/logadm
15 3 * * 0 /usr/lib/fs/nfs/nfsfind
1 2 * * * [ -x /usr/sbin/rtc ] && /usr/sbin/rtc -c > /dev/null 2>&1
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean
# crontab -l jones
13 13 * * * cp /home/jones/work_files /usr/backup/. > /dev/null 2>&1
```

crontab ファイルの削除

デフォルトでは、rm コマンドを使用して誤って crontab ファイルを削除してしまうことがないように、crontab ファイルは保護されています。crontab ファイルを削除する場合は、rm コマンドではなく crontab -r コマンドを使用してください。

デフォルトでは、crontab -r は、このコマンドを実行したユーザーの crontab を削除します。root またはその他のユーザーの crontab を削除するには、スーパーユーザーにならなければなりません。

このコマンドを使用するには、ディレクトリを (crontab ファイルが入っている) /var/spool/cron/crontabs に変更する必要はありません。

▼ crontab ファイルを削除する方法

1. (省略可能) root または他のユーザーの crontab ファイルを削除するには、スーパーユーザーになります。
2. 次のように入力して、crontab ファイルを削除します。

```
$ crontab -r [username]
```

username は、削除する crontab ファイルのユーザーのアカウント名を指定します。他のユーザーの crontab ファイルを削除するには、スーパーユーザーの権限が必要です。



注意 – 誤ってオプションを指定しないで crontab コマンドを入力した場合は、使用しているエディタの中断文字を入力してください。この文字を入力すると、変更結果を保存せずに crontab コマンドを終了できます。この場合に変更結果を保存してファイルを終了すると、既存の crontab ファイルが空のファイルで上書きされます。

3. crontab ファイルが削除されたことを確認します。

```
$ ls /var/spool/cron/crontabs
```

例 — crontab ファイルを削除する

次の例では、ユーザー smith が `crontab -r` コマンドを使用して自分の crontab ファイルを削除します。

```
$ ls /var/spool/cron/crontabs
adm    jones    lp      root    smith   sys     uucp
$ crontab -r
$ ls /var/spool/cron/crontabs
adm    jones    lp      root    sys     uucp
```

crontab コマンドの使用制御

`cron.deny` と `cron.allow` という `/etc/cron.d` ディレクトリ内の 2 つのファイルを使用して、`crontab` の使用を制御できます。これらのファイルによって、指定したユーザーだけが、それぞれ自分の `crontab` ファイルの作成、編集、表示、または削除などの `crontab` コマンドのタスクを実行できるようにします。

`cron.deny` および `cron.allow` ファイルは、それぞれ 1 行に 1 ユーザー名が入ったリストからなります。これらの使用制御用ファイルは、次のように連携して機能を果たします。

- `cron.allow` が存在する場合は、このファイルにリストされているユーザーだけが `crontab` ファイルを作成、編集、表示、または削除できます。
- `cron.allow` が存在しない場合は、`cron.deny` にリストされているユーザーを除くすべてのユーザーが `crontab` ファイルの実行を依頼できます。
- `cron.allow` も `cron.deny` も存在しない場合は、`root` 以外は `crontab` コマンドを実行できません。

`cron.deny` と `cron.allow` ファイルを編集または作成するには、スーパーユーザーの権限が必要です。

Solaris ソフトウェアのインストール時に作成される `cron.deny` ファイルには、次のユーザー名が含まれます。

```
$ cat /etc/cron.d/cron.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
```

デフォルトの `cron.deny` ファイル内のユーザー名は、いずれも `crontab` コマンドを使用できません。このファイルを編集して、`crontab` コマンドの使用を拒否したいユーザー名を追加できます。

デフォルトでは、`cron.allow` ファイルは提供されません。つまり、Solaris ソフトウェアをインストールした直後は、すべてのユーザー (デフォルトの `cron.deny` ファイルにリストされているユーザーを除く) が `crontab` コマンドを使用できます。`cron.allow` ファイルを作成した場合、そのユーザーだけが `crontab` コマンドを使用できます。

▼ `crontab` の使用を拒否する方法

1. スーパーユーザーになります。
2. `/etc/cron.d/cron.deny` ファイルを編集し、`crontab` コマンドを使用させないユーザー名を 1 行に 1 つずつ追加します。

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
username1
username2
username3
.
.
.
```

3. `/etc/cron.d/cron.deny` ファイルを確認します。

```
# cat /etc/cron.d/cron.deny
```

▼ `crontab` の使用を特定のユーザーに限定する方法

1. スーパーユーザーになります。
2. `/etc/cron.d/cron.allow` ファイルを作成します。
3. `root` ユーザー名を `cron.allow` ファイルに入力します。
このファイルに `root` を追加しないと、スーパーユーザーの `crontab` コマンドの使用が拒否されます。
4. 次のように、`crontab` コマンドの使用を許可するユーザー名を 1 行に 1 つずつ入力します。

```
root
username1
```

```
username2
username3
.
.
.
```

例 — crontab の使用を特定のユーザーに限定する

次は、ユーザー visitor、jones、temp に crontab を使用させない cron.deny ファイルの例です。

```
$ cat /etc/cron.d/cron.denydaemon
bin
smtp
nuucp
listen
nobody
noaccess
jones
temp
visitor
```

次は cron.allow ファイルの例です。ユーザー smith、jones、lp および root だけが、crontab コマンドを使用できます。

```
$ cat /etc/cron.d/cron.allow
root
jones
lp
smith
```

▼ crontab の使用制限を確認する方法

特定のユーザーが crontab を使用できるかどうかを確認するには、そのユーザーのアカウントでログインして crontab -l コマンドを使用します。

```
$ crontab -l
```

そのユーザーが crontab を使用できて、すでに crontab ファイルを作成している場合は、その crontab ファイルが表示されます。そのユーザーが crontab を使用できるが、crontab ファイルがない場合は、次のようなメッセージが表示されます。

```
crontab:crontab ファイルを開けません
```

このユーザーは、cron.allow (が存在する場合) に含まれているか、cron.deny に含まれていません。

ユーザーが crontab コマンドを使用できない場合は、上記の crontab ファイルの有無に関わらず、次のメッセージが表示されます。

```
crontab: cron を使用許可されていません
```

このメッセージは、ユーザーが `cron.allow` (が存在する場合) に含まれていないか、`cron.deny` ファイルに含まれていることを意味します。

1つのシステムタスクのスケジューリング (at)

以降の節では、`at` コマンドを使用して特定の時刻にジョブ (コマンドとスクリプト) をスケジューリングする方法、これらのジョブを表示して削除する方法、`at` コマンドの使用を制御する方法について説明します。

デフォルトでは、ユーザーはそれぞれ自分の `at` ジョブファイルを作成、表示、または削除できます。`root` または他のユーザーの `at` ファイルにアクセスするには、スーパーユーザーの権限が必要です。

`at` ジョブの実行を指定すると、`at` ジョブにジョブ識別番号と `.a` 拡張子が与えられ、それがジョブのファイル名になります。

at コマンドの説明

`at` ジョブの実行を指定するには、次の手順に従います。

1. コマンド実行時刻を指定して `at` ユーティリティを起動します。
2. 後で実行させるコマンドまたはスクリプトを入力します。

注 – このコマンドまたはスクリプトからの出力が重要な場合は、後で調べることができるように、出力内容を必ずファイルに書き込むようにしてください。

たとえば、次の `at` ジョブは、7月31日の真夜中に `smith` のユーザーアカウントから `core` ファイルを削除します。

```
$ at 11:45pm July 31
at> rm /home/smith/*core*
at> Press Control-d
commands will be executed using /bin/csh
job 933486300.a at Tue Jul 31 23:45:00 2001
```

at コマンドの使用制御

特定のユーザーだけがそれぞれの at ジョブに関する待ち行列情報を作成、削除、または表示できるように、at コマンドの使用を制御するファイルを設定できます。at コマンドの使用を制御するファイルは /etc/cron.d/at.deny です。ここにはユーザー名が列挙 (1 行に 1 人) されています。このファイルに列挙されているユーザーは、at コマンドを使用できません。

Solaris ソフトウェアのインストール時に作成される at.deny ファイルには、次のユーザー名が含まれます。

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
```

スーパーユーザーの特権があれば、at.deny ファイルを編集して、at の使用を制限したい他のユーザー名を追加できます。

▼ at ジョブを作成する方法

1. at ユーティリティを起動して、ジョブを実行したい時間を指定します。

```
$ at [-m] time [date]
```

-m	ジョブ終了後にメールを送る
time	ジョブをスケジュールしたい時刻の時単位の値。24 時間制を使用しない場合は、am または pm を追加する。使用できるキーワードは、midnight、noon、now。分単位の値の指定は省略可能
date	月または曜日の名前の最初の 3 英字以上、またはキーワード today または tomorrow

2. at プロンプトに、実行したいコマンドまたはスクリプトを 1 行に 1 つずつ入力します。
各行の終わりで Return キーを押すことにより、複数のコマンドを入力できます。
3. at ユーティリティを終了し、Control-D キーを押して at ジョブを保存します。
作成できた at ジョブは待ち行列番号を割り当てられ、それがそのジョブのファイル名にもなります。この番号は at ユーティリティの終了時に表示されます。

例 — at ジョブを作成する

次の例は、ユーザー `jones` が彼女のバックアップファイルを 7:30 pm に削除するように作成した at ジョブを示しています。彼女は、ジョブの終了後にメールメッセージを受け取れるように、`-m` オプションを使用しています。

```
$ at -m 1930
at> rm /home/jones/*.backup
at> Control-Dを押す
job 897355800.a at Thu Jul 12 19:30:00 2001
```

`jones` は次のメールメッセージを受け取りました。このメッセージは at ジョブが終了したことを確認しています。

```
Your "at" job "rm /home/jones/*.backup"
completed.
```

次の例は、`jones` が土曜の午前 4 時の大規模な at ジョブをどのようにスケジュールしたのかを示しています。ジョブの出力先は `big.file` です。

```
$ at 4 am Saturday
at> sort -r /usr/dict/words> /export/home/jones/big.file
```

▼ at 待ち行列を表示する方法

at 待ち行列で実行を待っているジョブを確認するには、次に示すように `atq` コマンドを使用します。このコマンドは、その使用者が作成した at ジョブに関する状態情報を表示します。

```
$ atq
```

▼ at ジョブを確認する方法

at ジョブが作成できたかどうかを確認するには、`atq` コマンドを使用します。次の `atq` コマンドは、`jones` に属する at ジョブが待ち行列に入っていることを確認しています。

```
$ atq
Rank      Execution Date      Owner      Job          Queue  Job Name
1st      Jul 12, 2001 19:30   jones     897355800.a   a      stdin
2nd      Jul 14, 2001 23:45   jones     897543900.a   a      stdin
3rd      Jul 17, 2001 04:00   jones     897732000.a   a      stdin
```

▼ at ジョブを表示する方法

自分の at ジョブの実行時刻に関する情報を表示するには、次のように `at -l` コマンドを使用します。

```
$ at -l [job-id]
```

-l *job-id* オプションで、状態を表示したいジョブの識別番号を指定します。

例 — at ジョブを表示する

次の例は、at -l コマンドからの出力を示しています。このコマンドは、特定のユーザーが依頼したすべてのジョブに関する状態を表示します。

```
$ at -l
897543900.a    Sat Jul 14 23:45:00 2001
897355800.a    Thu Jul 12 19:30:00 2001
897732000.a    Tue Jul 17 04:00:00 2001
```

次の例は、at -l コマンドに1つのジョブを指定して表示された出力を示しています。

```
$ at -l 897732000.a
897732000.a    Tue Jul 17 04:00:00 2001
```

▼ at ジョブを削除する方法

1. (省略可能) root または他のユーザーの at ジョブを削除する場合は、スーパーユーザーになります。
2. 次のように入力して、at ジョブが実行される前に待ち行列から削除します。

```
$ at -r [job-id]
```

-r *job-id* オプションで、削除したいジョブの識別番号を指定します。

3. at -l (または atq) コマンドを使用して、at ジョブが削除されていることを確認します。

at -l コマンドは、at 待ち行列に残っているジョブを表示します。識別番号を指定したジョブは、このリストに表示されないはずですが。

```
$ at -l [job-id]
```

例 — at ジョブを削除する

次の例では、ユーザーが7月17日の午前4時に実行されるようにスケジュールした at ジョブを削除しようとしています。まず、このユーザーは at 待ち行列を表示してそのジョブの識別番号を探します。次に、そのジョブを at 待ち行列から削除します。最後に、at 待ち行列をもう一度表示して上記のジョブが削除されていることを確認します。

```
$ at -l
897543900.a    Sat Jul 14 23:45:00 2001
897355800.a    Thu Jul 12 19:30:00 2001
897732000.a    Tue Jul 17 04:00:00 2001
```



```
$ at -r 897732000.a
$ at -l 897732000.a
at: 858142000.a: No such file or directory
```

▼ at コマンドの使用を拒否する方法

1. スーパーユーザーになります。
2. /etc/cron.d/at.deny ファイルを開きます。at コマンドを使用させないようにするユーザー名を 1 行に 1 つずつ追加または削除します。

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
username1
username2
username3
.
.
.
```

例 — at の使用を拒否する

次は、ユーザー smith と jones が at コマンドを使用できないように編集された at.deny ファイルの例です。

```
$ cat at.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
jones
smith
```

▼ at の使用の拒否を確認する方法

特定のユーザー名が正しく /etc/cron.d/at.deny に追加されているかどうかを確認するには、そのユーザー名でログインして、at -l コマンドを使用します。そのユーザーが at コマンドを使用できない場合は、次のメッセージが表示されます。

```
# su smith
Password:
$ at -l
```

at: 使用許可されていません

同様に、そのユーザーが at ジョブの実行を依頼しようとした場合は、次のメッセージが表示されます。

```
$ at 2:30pm
```

at: 使用許可されていません

このメッセージによって、そのユーザーが at.deny ファイルに含まれていることが確認されます。

at コマンドを使用できる場合、at -l コマンドは何も返しません。

第 19 章

システムアカウンティングの設定と管理 (手順)

この章では、システムアカウンティングを設定して維持する方法について説明します。

この章で説明する手順は次のとおりです。

- 281 ページの「システムアカウンティングを設定する方法」
- 283 ページの「ユーザーに課金する方法」
- 284 ページの「壊れた `wtmpx` ファイルの修復方法
- 285 ページの「`tacct` エラーを修復する方法」
- 286 ページの「`runacct` スクリプトを再起動する方法」
- 281 ページの「システムアカウンティングを設定する方法」
- 287 ページの「システムアカウンティングを永続的に無効にする方法」

この章の内容は以下のとおりです。

- 275 ページの「システムアカウンティング」
- 280 ページの「システムアカウンティングの設定」

拡張アカウンティング機能の使い方については、『Solaris のシステム管理 (資源管理とネットワークサービス)』の「拡張アカウンティング」を参照してください。

システムアカウントのさまざまなレポートについては、第 20 章を参照してください。

システムアカウンティング

SunOS 5.9 のシステムアカウンティングソフトウェアは、ユーザー接続時間、プロセスに使用された CPU 時間、およびディスク使用量についてのデータを収集および記録できるプログラム群です。一度このデータを収集すると、レポートを生成したり、システム使用に対して課金したりすることができます。

アカウントティングプログラムを使用すると、次の事項を行うことができます。

- システムの使用状況の監視
- パフォーマンス上の問題の追跡と解決
- システムセキュリティの維持

システムアカウントティングプログラムは、設定が済むと、ほとんどの場合自動的に実行されます。

システムアカウントティングの動作

自動アカウントティングは、まずアカウントティング起動スクリプトをルートの `crontab` ファイルに配置することによって設定します。すると、アカウントティング起動スクリプトが、`cron` コマンドによって自動的に起動されます。

次の概要は、システムアカウントティングのプロセスを示したものです。

1. システムを起動してからシャットダウンするまでの間に、システムの利用に関する (ユーザーログイン、実行されたプロセス、データの格納などの) raw データがアカウントティングファイルに収集されます。
2. 定期的に (通常 1 日に 1 回)、`/usr/lib/acct/runacct` スクリプトが各種のアカウントティングファイルを処理して、累積要約ファイルと日次アカウントングレポートを生成します。次に、`/usr/lib/acct/prdaily` スクリプトが日次レポートを印刷します。

`runacct` スクリプトについては、289 ページの「`runacct` スクリプト」を参照してください。
3. 毎月、`monacct` スクリプトを実行することによって、`runacct` 累積要約ファイルを処理して印刷します。`monacct` スクリプトによって生成される要約レポートは、月次またはその他の会計期間ベースのユーザーに対する効率的な課金手段になります。

システムアカウントティングのコンポーネント

アカウントティングユーティリティは、データから要約ファイルとレポートを生成する C 言語プログラムとシェルスクリプトを提供します。これらのプログラムは `/usr/lib/acct` ディレクトリにあります。アカウントングレポートは、`/var/adm/acct` ディレクトリにあります。

日次アカウントティングによって、次の 4 種類の監査を行うことができます。

- 接続
- プロセス
- ディスク
- 料金計算

接続アカウントティング

接続アカウントティングでは、次のデータを調べることができます。

- 特定のユーザーがログインしていた時間
- tty 回線の利用状況
- システムのリポート回数
- アカウントティングソフトウェアが有効または無効に設定された頻度

この情報を提供するために、システムは次のデータを格納します。

- 時間調節の記録
- ブート時刻
- アカウントティングソフトウェアが有効または無効に設定された時刻
- 実行レベルの変更
- ユーザープロセスの作成 (login プロセスと init プロセス)、およびプロセスの終了

これらのレコード (date、init、login、ttypmon、acctwtmp などのシステムプログラム出力によって生成されます) は /var/adm/wtmpx ファイルに格納されます。wtmpx ファイルのエントリには、次の情報を入れることができます。

- ログイン名
- 装置名
- プロセス ID
- エントリタイプ
- エントリがいつ作成されたのかを示すタイムスタンプ

プロセスアカウントティング

プロセスアカウントティングでは、システムで実行される各プロセスに関する次のようなデータを追跡できます。

- プロセスを使用するユーザーおよびグループのそれぞれのユーザー ID とグループ ID
- プロセスの開始時刻と経過時間
- プロセスの CPU 時間 (ユーザー時間とシステム時間)
- 使用されるメモリーの量
- 実行されるコマンド
- プロセスを制御する tty

プロセスが終了するたびに、exit プログラムは上記のデータを収集し、/var/adm/pacct ファイルに書き込みます。

ディスクアカウントティング

ディスクアカウントティングでは、各ユーザーがディスク上にもっているファイルについて次のデータを収集し、それらを整形できます。

- ユーザーのユーザー名とユーザー ID

■ ユーザーのファイルが使用しているブロック数

これらのデータはシェルスクリプト `/usr/lib/acct/dodisk` によって収集されますが、収集周期は `/var/spool/cron/crontabs/root` ファイルに追加する `cron` コマンドによって決定されます。一方、`dodisk` スクリプトは、`acctdisk` および `acctdusg` コマンドを起動して、ログインによるディスク使用量を収集させます。



注意 – `dodisk` スクリプトを実行して収集された情報は、`/var/adm/acct/nite/disktacct` ファイルに格納されます。これらの情報は、次に `dodisk` スクリプトを実行したときに上書きされます。したがって、`dodisk` スクリプトは同じ日に 2 回以上実行しないでください。

`acctdusg` コマンドは、ランダムに書き込まれたため穴があいたファイルに対して過剰に課金します。このような問題が起こるのは、`acctdusg` コマンドが、ファイルサイズを決めるときに、ファイルの間接ブロックを読み取らないからです。`acctdusg` コマンドは、`i` ノードの `di_size` の値をチェックして、ファイルのサイズを決めます。

ユーザー料金の計算

`chargefee` ユーティリティは、ユーザーに提供した特別なサービス (ファイルの復元など) に対する課金を、`/var/adm/fee` ファイルに格納します。このファイルの各エントリは、ユーザーのログイン名、ユーザー ID、および料金から構成されています。このファイルは、`runacct` スクリプトによって毎日チェックされて、新しいエントリが全体のアカウントレコードにマージされます。`chargefee` スクリプトを実行してユーザーに課金する命令については、283 ページの「ユーザーに課金する方法」を参照してください。

日次アカウント機能の動作

次に、SunOS の日次アカウント機能がどのように動作するかを要約して示します。

1. システムをマルチユーザーモードに切り替えると、`/usr/lib/acct/startup` プログラムが実行されます。この `startup` プログラムは、それぞれ日次アカウント機能を呼び出す他のプログラムを実行します。
2. `acctwtmp` プログラムは `/var/adm/wtmpx` ファイルに「ブート」レコードを追加します。このレコードには、システム名が `wtmpx` レコード内のユーザー名として示されます。次の表に、`raw` アカウントデータがどのように収集され、どこに格納されるかをまとめて示します。

表 19-1 raw アカウンティングデータ

/var/adm 内のファイル	格納される情報	データを書き込むプログラム	表記形式
wtmpx	接続セッション数	login, init	バイナリ
	日付変更	date	
	リポート	acctwtmp	
	シャットダウン	shutacct	
pacctn	プロセス	カーネル(プロセス終了時) turnacct switch (古いファイルの内容が 500 ブロックに達すると、新しいファイルが作成される)	バイナリ
fee	特別料金	chargefee	ASCII
acct/nite/diskacct	使用ディスク領域	dodisk	バイナリ

- turnacct スクリプトが `-on` オプションで起動されて、プロセスアカウンティングを開始します。turnacct スクリプトは、特に `/var/adm/pacct` 引数を使用して `accton` プログラムを実行します。
- remove シェルスクリプトが、runacct によって `sum` ディレクトリに保存されている `pacct` および `wtmpx` ファイルを「整理」します。
- login および `init` プログラムが、`/var/adm/wtmpx` ファイルにレコードを書き込み、接続セッションを記録します。すべての日付変更(引数を指定して `date` を使用)も `/var/adm/wtmpx` ファイルに書き込まれます。リポート回数とシャットダウン回数も、`acctwtmp` コマンドを使用して、`/var/adm/wtmpx` ファイルに記録されます。
- プロセスが終了すると、カーネルが `/var/adm/pacct` ファイルにプロセスごとに 1 レコードを `acct.h` 形式で書き込みます。
cron コマンドは、1 時間ごとに `ckpacct` スクリプトを実行して、`/var/adm/pacct` ファイルのサイズをチェックします。このファイルが 500 ブロック(デフォルト)よりも大きくなった場合は、`turnacct switch` コマンドが実行されます。(このプログラムは `pacct` ファイルを `pacctn` ファイルに移動して、新しいファイルを作成します。) `pacct` ファイルを小さく分けることの利点は、それらのレコードを処理するときに障害が発生して、`runacct` スクリプトを再起動しようとしたときに明らかになります。
- runacct スクリプトは、cron コマンドによって毎晩実行されます。runacct スクリプトは、`/var/adm/pacctn`、`/var/adm/wtmpx`、`/var/adm/fee` および `/var/adm/acct/nite/diskacct` のアカウンティングファイルを処理し、ユーザー別のコマンド要約と利用状況要約を生成します。

8. /usr/lib/acct/prdaily スクリプトは、runacct スクリプトによって1日1回実行され、/var/adm/acct/sum/rprt MMDD ファイルに日次アカウントリング情報を書き込みます。
9. monacct スクリプトが月に1回(または毎会計期の終わりなど、ユーザーが決めた周期で)実行されます。monacct スクリプトは、sum ディレクトリに格納されているデータに基づいてレポートを作成します。これらのデータは runacct スクリプトによって毎日更新されています。このレポートを作成後、monacct スクリプトは sum ディレクトリを「整理」して、新しい runacct データを格納するためのファイルを準備します。

システムがシャットダウンしたときの動作

shutdown コマンドを使用してシステムをシャットダウンした場合は、shutacct スクリプトが自動的に実行されます。shutacct スクリプトは /var/adm/wtmpx ファイルに理由レコードを書き、アカウントリングプロセスを無効に設定します。

システムアカウントリングの設定

システムアカウントリングは、システムがマルチユーザーモード (システム状態 2) のときに実行されるように設定できます。システムアカウントリングには、次の内容が含まれます。

1. /etc/rc0.d/K22acct および /etc/rc2.d/S22acct 起動スクリプトの作成
2. crontab ファイル /var/spool/cron/crontabs/adm および /var/spool/cron/crontabs/root の変更

表 19-2 に、デフォルトのアカウントリング管理スクリプトを説明します。

表 19-2 デフォルトのアカウントリング管理スクリプト

アカウントリングスクリプト	マニュアルページ	種類	実行頻度
ckpacct	ckpacct (1M)	/usr/adm/pacct ログファイルのサイズをチェックし、ファイルが大きくなりすぎていないことを確認する	定期的

表 19-2 デフォルトのアカウント管理スクリプト (続き)

アカウント管理スクリプト	マニュアルページ	種類	実行頻度
runacct	runacct (1M)	接続、ディスク、および料金のアカウント管理情報を処理する。このスクリプトから、処理したくないアカウント管理機能のコマンドを削除できる	日次
monacct	monacct (1M)	会計アカウント管理要約レポートを月に1回のペースで生成する。このスクリプトの実行頻度を定めることができる。このスクリプトから、生成したくないアカウント管理機能のコマンドを削除できる	会計期間に基づく

デフォルトでどのアカウント管理スクリプトを実行するかを選択できます。これらのエントリを `crontab` ファイルに追加すると、アカウント管理は自動的に実行されるようになります。

▼ システムアカウント管理を設定する方法

1. スーパーユーザーになります。
2. 必要な場合は、`pkgadd` コマンドを使用して、システムに `SUNWacctr` と `SUNWaccu` パッケージをインストールします。
3. 次のように入力して、`/etc/init.d/acct` を実行レベル 2 の起動スクリプトとしてインストールします。

```
# ln /etc/init.d/acct /etc/rc2.d/S22acct
```
4. `/etc/init.d/acct` を実行レベル 0 の停止スクリプトとしてインストールします。

```
# ln /etc/init.d/acct /etc/rc2.d/S22acct
```
5. `ckpacct`、`runacct`、および `monacct` スクリプトが自動的に起動するように、`adm` ユーザーの `crontab` ファイルに次の行を追加します。

```
# EDITOR=vi; export EDITOR
# crontab -e adm
0 * * * * /usr/lib/acct/ckpacct
30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
30 7 1 * * /usr/lib/acct/monacct
```
6. `dodisk` スクリプトが自動的に起動するように、`root` の `crontab` ファイルに次の行を追加します。

```
# crontab -e
30 22 * * 4 /usr/lib/acct/dodisk
```

7. /etc/acct/holidays を編集して祭日と休日を含めます。
詳細は、holidays(4) のマニュアルページおよび次の例を参照してください。
8. システムをリブートするか、手入力でアカウントिंगを起動します。

```
# /etc/init.d/acct start
```

例 — アカウンティングを設定する

この編集された adm の crontab には、ckpacct、runacct および monacct スクリプトが含まれています。

```
#ident "@(#)adm 1.5 92/07/14 SMI" /* SVr4.0 1.2 */
#
# The adm crontab file should contain startup of performance
# collection if the profiling and performance feature has been
# installed.
0 * * * * /usr/lib/acct/ckpacct
30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
30 7 1 * * /usr/lib/acct/monacct
```

この編集された root の crontab には、dodisk プログラムのエントリが含まれています。

```
#ident "@(#)root 1.19 98/07/06 SMI" /* SVr4.0 1.1.3.1 */
#
# The root crontab should be used to perform accounting data collection.
#
# The rtc command is run to adjust the real time clock if and when
# daylight savings time changes.
#
10 3 * * * /usr/sbin/logadm
15 3 * * 0 /usr/lib/fs/nfs/nfsfind
1 2 * * * [ -x /usr/sbin/rtc ] && /usr/sbin/rtc -c> /dev/null 2>&1
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean
30 22 * * 4 /usr/lib/acct/dodisk
```

次に /etc/acct/holidays ファイルの例を示します。

```
* @(#)holidays January 1, 2001
*
* Prime/Nonprime Table for UNIX Accounting System
*
* Curr Prime Non-Prime
* Year Start Start
*
* 1999 0800 1800
*
* only the first column (month/day) is significant.
*
```

アカウント情報の管理

この節では、壊れたアカウントファイルを修復し、runacct スクリプトを再起動する方法を説明します。

壊れたファイルと wtmpx エラーを修復する

アカウントシステムは絶対に障害を発生しないというものではありません。ファイルが壊れたり、失われることがあります。ファイルによっては、単に無視してよいものや、バックアップから復元できるものがあります。ただし、特定のファイルは、アカウントシステムの完全性を維持するために修復しなければなりません。

wtmpx ファイルは、アカウントシステムを日常的に運用する上で発生する問題の大部分の原因になっています。日付を手動で変更した時に、システムがマルチユーザーモードの場合は、日付変更レコードが/var/adm/wtmpx ファイルに書き込まれます。wtmfix ユーティリティは、日付変更が行われたときに、wtmp レコードのタイムスタンプを調整するように設計されています。ただし、日付変更とリポートの組み合わせによっては、wtmfix の処理から漏れて、acctcon プログラムの処理が失敗することがあります。

▼ 壊れた wtmpx ファイルの修復方法

1. スーパーユーザーになります。
2. /var/adm ディレクトリに移動します。
3. wtmpx ファイルの形式をバイナリから **ASCII** に変換します。
4. wtmpx.ascii を編集して、壊れたレコードを削除します。
5. wtmpx.ascii ファイルを再びバイナリファイルに変換します。

```
# /usr/lib/acct/fwtmp < wtmpx > wtmpx.ascii
```

```
# /usr/lib/acct/fwtmp -ic < wtmpx.ascii> wtmpx
```

詳細は、fwtmp(1M) のマニュアルページを参照してください。

tacct エラーを修復する

/var/adm/acct/sum/tacct ファイルの完全性は、システム資源に対してユーザーに課金している場合は重要です。不正な tacct レコードが、負の数値、重複ユーザー ID、65535 のユーザー ID で表示されることがあります。このような場合はま

ず、prtacct スクリプトを使用して/var/adm/acct/sum/tacctprev ファイルを印刷してチェックします。内容が正しい場合は、最新の/var/adm/acct/sum/tacctMMDD ファイルを使用して、/var/adm/acct/sum/tacct ファイルを作成し直してください。次の手順は、簡単な修復手順の概要を説明しています。

▼ tacct エラーを修復する方法

1. スーパーユーザーになります。
2. ディレクトリ /var/adm/acct/sum に移動します。
3. tacctMMDD ファイルの形式をバイナリから **ASCII** に変換します。

```
# /usr/lib/acct/acctmerg -v < tacctMMDD > xtacct
```

MMDD は、月と日をあらわす 2 桁の数値を組み合わせたものです。
4. xtacct ファイルを編集して、壊れたレコードを削除し、重複レコードを別のファイルに書き込みます。
5. xtacct ファイルを **ASCII** 形式からバイナリに変換します。

```
# /usr/lib/acct/acctmerg -i < xtacct > tacctMMDD
```

MMDD は、月と日をあらわす 2 桁の数値を組み合わせたものです。
6. ファイル tacctprev と tacctMMDD をマージしてファイル tacct を生成します。

```
# /usr/lib/acct/acctmerg < tacctprev tacctMMDD > tacct
```

runacct スクリプトを再起動する

runacct スクリプトは、さまざまな理由で失敗することがあります。最も一般的な理由には、システムクラッシュ、/var ディレクトリの容量不足での実行、wtmptx ファイルが壊れたなどがあります。activeMMDD ファイルが存在する場合は、まずエラーメッセージをチェックします。active ファイルおよび lock ファイルが存在する場合は、異常なメッセージがないかどうか fd2log ファイルを調べます。

runacct スクリプトは、引数を指定しないで実行すると、その実行がその日の最初の実行であるとみなします。runacct スクリプトを再起動し、runacct スクリプトがアカウントを返す月と日を指定する場合は、引数 MMDD が必要になります。処理のエントリーポイントは statefile ファイルの内容に基づきます。statefile ファイルを無効にするには、処理を開始したい状態をコマンド行に指定します。利用可能な状態の説明については、runacct (1M) のマニュアルページを参照してください。



注意 - runacct プログラムを手動で実行するときは、ユーザー adm として実行していることを確認してください。

▼ runacct スクリプトを再起動する方法

1. lastdate ファイルと lock* ファイル (もしあれば) を削除します。

```
$ cd /var/adm/acct/nite
$ rm lastdate lock*
```

lastdate ファイルには、runacct プログラムが最後に実行された日付が含まれています。次の手順で runacct を再起動することによって、このファイルが再び作成されます。

2. runacct スクリプトを再起動します。

```
$ /usr/lib/acct/runacct MMDD [state] 2> /var/adm/acct/nite/fd2log &
```

MMDD

月日を2桁の数値で指定する

state

runacct 処理を開始させたい状態または開始点を指定する。

システムアカウントの停止と無効

システムアカウントは、一時的に停止することも、永続的に無効にすることもできます。

▼ 一時的にシステムアカウントを停止する方法

1. スーパーユーザーになります。
2. 適切な行をコメントアウトすることによって、プログラム ckpacct、runacct、および monacct の実行が停止するように、ユーザー adm の crontab ファイルを編集します。

```
# EDITOR=vi; export EDITOR
# crontab -e adm
#0 * * * * /usr/lib/acct/ckpacct
```

```
#30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
#30 7 1 * * /usr/lib/acct/monacct
```

3. 適切な行をコメントアウトすることによって、プログラム `dodisk` の実行が停止するように、ユーザー `root` の `crontab` ファイルを編集します。

```
# crontab -e
#30 22 * * 4 /usr/lib/acct/dodisk
```

4. 次のように入力して、アカウントングプログラムを停止します。

```
# /etc/init.d/acct stop
```

5. (省略可能)新規に追加したコメント記号を `crontab` ファイルから削除し、アカウントングプログラムを再起動して、システムアカウントングを再び利用できるようにします。

```
# /etc/init.d/acct start
```

▼ システムアカウントングを永続的に無効にする方法

1. スーパーユーザーになります。
2. ユーザー `adm` の `crontab` ファイルを編集して、プログラム `ckpacct`、`runacct`、および `monacct` 用のエントリを削除します。

```
# EDITOR=vi; export EDITOR
# crontab -e adm
```

3. ユーザー `root` の `crontab` ファイルを編集して、プログラム `dodisk` 用のエントリを削除します。

```
# crontab -e
```

4. 実行レベル `2` 用の起動スクリプトのリンクを取り外します。

```
# unlink /etc/rc2.d/S22acct
```

5. 実行レベル `0` 用の停止スクリプトのリンクを取り外します。

```
# unlink /etc/rc0.d/K22acct
```

6. 次のように入力して、アカウントングプログラムを停止します。

```
# /etc/init.d/acct stop
```


第 20 章

システムアカウントティング (リファレンス)

この章では、システムアカウントの参照情報について説明します。

この章の内容は、次のとおりです。

- 292 ページの「日次アカウントングレポート」
- 289 ページの「runacct スクリプト」
- 299 ページの「システムアカウントングファイル」

runacct スクリプト

主となる日次アカウントングスクリプトである runacct は、通常はプライムタイムつまり最も忙しい時間帯を避けて cron コマンドにより起動されます。この runacct スクリプトは、接続、料金、ディスク、プロセス用の各アカウントングファイル进行处理します。さらに、このスクリプトは、課金目的で prdaily と monacct スクリプトが使用する日次および累積要約ファイルも準備します。

runacct スクリプトは、エラーが発生した場合、ファイルを壊さないよう注意します。一連の保護機構により、エラーを認識し、インテリジェント診断を提供し、最小のユーザー介入で runacct スクリプトを起動し直せるように処理を完了しようとしています。runacct は、説明メッセージを active というファイルに書き込み、進捗状況を記録します。runacct スクリプトが使用するファイルは、特記しない限り、/var/adm/acct/nite ディレクトリにあります。runacct スクリプト実行中の診断内容の出力は、fd2log ファイルに書き込まれます。

runacct スクリプトが実行されると、lock および lock1 ファイルが作成されます。これらのファイルは、runacct が同時に実行されるのを防ぎます。runacct プログラムは、起動時にこれらのファイルが存在すれば、エラーメッセージを出力します。lastdate ファイルには runacct スクリプトが最後に起動されたときの月日が入っており、このファイルは runacct スクリプトを 1 日に 1 回しか実行できないようにするために使用されます。runacct スクリプトがエラーを検出した場合は、エ

ラーメッセージがコンソールに出力され、メールが root と adm に送られ、ロックが解除され、診断ファイルが保存され、実行が終了されます。runacct スクリプトを再び起動する手順については、286 ページの「runacct スクリプトを再起動する方法」を参照してください。

runacct スクリプトを再起動可能にするために、処理は再入可能な別々の状態に分割されます。statefile というファイルを使用して、最後に終了した状態が追跡されます。各状態が終了するたびに、statefile は更新されて次の状態に反映されます。1つの状態の処理が終了すると、statefile ファイルが読み取られて次の状態が処理されます。runacct スクリプトは、CLEANUP 状態に達すると、ロックを解除して実行を終了します。各状態は次の表のように実行されます。

表 20-1 runacct 状態

状態	説明
SETUP	turnacct switch コマンドが実行されて新しい pacct ファイルが作成される。/var/adm/pacctn プロセスアカウントングファイル (pacct ファイルを除く) は、/var/adm/Spacctn.MMDD ファイルに移される。/var/adm/wtmpx ファイルは (最後に現在の時のレコードが追加されて) /var/adm/acct/nite/wtmp.MMDD に移され、新しい /var/adm/wtmp が作成される。closewtmp および utmp2wtmp プログラムは、現在ログインしているユーザーに課金するために wtmp.MMDD ファイルと新しい wtmpx ファイルにレコードを追加する。
WTMPFIX	wtmpfix プログラムが nite ディレクトリ内の wtmp.MMDD ファイルを調べて誤りがないか確認する。日付の変更により、acctcon プログラムの失敗が引き起こされることがあるため、日付のレコードが変更されると、wtmpfix プログラムは wtmpx ファイルのタイムスタンプを調整しようとする。さらに、wtmpx ファイルからのエントリが壊れていた場合、壊れたエントリをすべて削除する。問題が解決された後のバージョンの wtmp.MMDD ファイルは、tmpwtmp ファイルに書き込まれる。
CONNECT	acctcon プログラムが使用されて、ctacct.MMDD ファイルに接続アカウントングレコードが記録される。これらのレコードは tacct.h 形式になっている。acctcon は、さらに lineuse および reboots ファイルを作成する。reboots ファイルは、wtmpx ファイルで見つかったすべてのブートレコードを記録する
PROCESS	acctprc プログラムが使用されて、/var/adm/Spacctn.MMDD プロセス課金ファイルを ptacctn.MMDD ファイルのアカウントングレコードの合計に変換する。runacct スクリプトが失敗した場合に、Spacct ファイルが処理されないように、Spacct ファイルと ptacct ファイルは番号で照合される
MERGE	acctmerg プログラムが、プロセスアカウントングレコードを接続アカウントングレコードとマージして daytacct ファイルを作成する

表 20-1 runacct 状態 (続き)

状態	説明
FEES	acctmerg プログラムが、fee ファイルからの ASCII tacct レコードを daytacct ファイルにマージする
DISK	dodisk スクリプトの手続きが実行されて disktacct ファイルが作成されている場合は、そのファイルを DISK プログラムは daytacct ファイルにマージし、diskacct ファイルを /tmp/diskacct.MMDD ファイルに移動させる
MERGETACCT	acctmerg プログラムが、daytacct を累積合計課金ファイルである sum/tacct ファイルとマージする。毎日 daytacct ファイルが sum/tacct.MMDD に保存されるため、sum/tacct が壊れたり失われたりしても、作成し直すことができる
CMS	acctcms プログラムが数回実行される。まず、このプログラムが実行され、Spacctn ファイルを使用してコマンド要約を生成し、sum/daycms ファイルに日付を書き込む。次に、acctcms プログラムは、sum/daycms ファイルと sum/cms 累積コマンド要約ファイルをマージする。最後に、acctcms プログラムは、nite/daycms および nite/cms という ASCII コマンド要約ファイルを、それぞれ sum/daycms および sum/cms ファイルから作成する。lastlogin プログラムを使用してログファイル /var/adm/acct/sum/loginlog が作成される。これは、各ユーザーが最後にログインした時刻を示すレポートである。runacct スクリプトが真夜中を過ぎてから実行された場合は、いずれかのユーザーが最後にログインした時刻を示す日付が 1 日狂うことになる
USEREXIT	インストールに依存する任意の (ローカル) 課金プログラムをこの時点で取り入れることができる。runacct スクリプトは、そのプログラムを /usr/lib/acct/runacct.local プログラムと呼ばれるものと想定する
CLEANUP	一時ファイルが整理され、prdaily スクリプトが実行され、その出力が sum/rpt.MMDD ファイルに保存され、ロックが解除されて終了する



注意 - runacct スクリプトを CLEANUP 状態で起動し直すときは、最後の ptacct ファイルが不完全であるため、このファイルを削除してください。

日次アカウンティングレポート

runacct シェルスクリプトは、呼び出されるたびに基本的な 5 種類のレポートを生成します。5 つの基本レポートは次のとおりです。

表 20-2 日次アカウンティングレポート

レポートの種類	説明
292 ページの「日次レポート」	tty 番号別の端末回線の利用状況を示す
293 ページの「日次利用状況レポート」	ユーザー別のシステム資源の利用状況を示す。ユーザー ID 順に表示される
295 ページの「日次コマンド要約」	コマンド別のシステム資源の利用状況を示す。メモリー使用量が大きい順に表示される。つまり、メモリーを最も多く使用したコマンドから先に表示される。これと同じ情報が月次コマンド概要で該当月について報告される
296 ページの「月次コマンド要約」	monacct プログラムの最新の実行から累積した日付を反映した累積概要
297 ページの「最終ログインレポート」	各ユーザーが最後にログインした日付を示す。日付順に表示される

日次レポート

このレポートは、使用された各端末回線に関する情報を示します。次に日次レポートの例を示します。

```
Oct 16 02:30 2001 DAILY REPORT FOR venus Page 1
```

```
from Mon Oct 15 02:30:02 2001
to   Tue Oct 16 02:30:01 2001
1    runacct
1    acctcon
```

```
TOTAL DURATION IS 1440 MINUTES
```

```
LINE      MINUTES  PERCENT  # SESS  # ON  # OFF
console   868      60       1      1    2
TOTALS    868      --       1      1    2
```

from および to の行は、レポートに反映される時間帯を指定します。この時間帯とは、直前の日次レポートが生成された時間から現在の日次レポートが生成されるまでの時間のことです。次に、システムリブートのログ、シャットダウン、電源異常からの回復、acctwtmp プログラムによって /var/adm/wtmpx ファイルにダンプされたその他のレコードが続きます。詳細は、acct (1M) のマニュアルページを参照してください。

このレポートの第 2 部は回線利用状況の内訳です。TOTAL DURATION は、システムがどれだけの時間マルチユーザーモード (端末回線を通してアクセス可能です) であったかを示します。次の表は、日次レポートのデータを説明したものです。

表 20-3 日次レポート

列	説明
LINE	端末回線またはアクセスポート
MINUTES	アカウント期間を通じてこの回線が使用中であった合計分
PERCENT	TOTAL DURATION を MINUTES の合計値で割った数値
# SESS	この回線またはポートがログインセッション用にアクセスされた回数
# ON	SESS に同じ。(このカラムにはそれ以上の意味はない。回線またはポートがユーザーのログインに使用された回数を表示する)
# OFF	このカラムは、この回線でユーザーがログアウトした回数と発生した割り込みを表す。割り込みは一般にシステムがマルチユーザーモードにされてから ttymon が初めて起動されたときに発生する。# OFF が大きな割合で # SESS を上回る場合は、マルチプレクサ、モデム、ケーブルに障害があるか、どこかに接続の問題がある可能性がある。一番考えられる原因は、マルチプレクサからのケーブルの接続が外れたままになっていることである

マシンの稼動中は、/var/adm/wtmpx ファイルが接続アカウントの元になるため、このファイルを監視する必要があります。wtmpx ファイルが急速に大きくなる場合は、次のコマンドを実行して、どの tty 回線が最も使用頻度が高いかを調べてください。

```
# /usr/lib/acct/acctcon -l file < /var/adm/wtmpx
```

割り込みが頻繁に発生する場合は、全般的なシステムのパフォーマンスが影響を受けることになります。さらに、wtmp ファイルが壊れることもあります。この問題を解決するには、284 ページの「壊れた wtmpx ファイルの修復方法」を参照してください。

日次利用状況レポート

このレポートは、システム資源の利用状況のユーザー別の内訳を示します。このレポートの例は、次のとおりです。

UID	LOGIN NAME	CPU PRIME	(MINS) NPRIME	KCORE- PRIME	MINS NPRIME	CONNECT PRIME	(MINS) NPRIME	DISK BLOCKS	# OF PROCS	# OF SESS	# DISK SAMPLES	FEE
0	TOTAL	72	148	11006173	51168	26230634	57792	539	330	0	2150	1
0	root	32	76	11006164	33664	26230616	22784	0	0	0	127	0
4	adm	0	0	22	51	0	0	0	420	0	0	0
101	rimmer	39	72	894385	1766020	539	330	0	1603	1	0	0

次の表は、日次利用状況レポートのデータを説明したものです。

表 20-4 日次利用状況レポート

列	説明
UID	ユーザー ID 番号
LOGIN NAME	ユーザーのログイン (またはユーザー) 名。複数のログイン名をもつユーザーを識別する
CPU (MINS)	ユーザーのプロセスが CPU を使用した時間を表す。このカテゴリの情報は、PRIME (プライムタイム時間帯) と NPRIME (プライムタイム時間帯外) に分けられる。アカウントシステムのこれらのデータのバージョンは、/etc/acct/holidays ファイルに格納されている
KCORE-MINS	プロセスが実行中に使用する累積メモリー量を表す。毎分あたりに使用される K バイトメモリーセグメント数を表す。この計量値も PRIME と NPRIME に分けられる
CONNECT (MINS)	ユーザーがシステムにログインしていた時間を分単位で表す。「実時間」とも呼ぶ。PRIME と NPRIME に分けられる。たとえば、この時間の値が大きく # OF PROCS の数値が小さい場合は、ログインの所有者がまず朝にログインし、その後はその日の終わりまで端末にほとんど触れていないと考えられる
DISK BLOCKS	acctdusg プログラムからの出力であり、ディスクアカウントプログラムを実行し、アカウントレコード (daytacct) をマージする。アカウントの目的では、ブロックは 512 バイト
# OF PROCS	ユーザーが起動したプロセス数を表す。数値が大きい場合は、ユーザーのシェルプロシージャが制御できなくなった可能性がある
# OF SESS	ユーザーがシステムにログインした回数
# DISK SAMPLES	平均ディスクブロック数 (DISK BLOCKS) を得るためにディスクアカウントが何回実行されたかを示す
FEE	chargefee スクリプトによってユーザーに課金された累積合計額を表す。使用されない場合が多い

日次コマンド要約

このレポートはコマンド別のシステム資源の利用状況を示します。このレポートでは、最も使用率の高いコマンドがわかり、それらコマンドがどのようにシステム資源を利用しているかに基づいて、どのようにしたらシステムの最適チューニングが可能かを知ることができます。

これらのレポートは TOTAL KCOREMIN によってソートされます。TOTAL KCOREMIN は任意の基準ですが、システムでのドレーンの計算にはすぐれた指標です。

日次コマンド要約の例を、次に示します。

TOTAL COMMAND SUMMARY									
COMMAND NAME	NUMBER CMDS	TOTAL KCOREMIN	TOTAL CPU-MIN	TOTAL REAL-MIN	MEAN SIZE-K	MEAN CPU-MIN	HOG FACTOR	CHARS TRNSFD	BLOCKS READ
TOTALS	2150	1334999.75	219.59	724258.50	6079.48	0.10	0.00	397338982	419448
netscape	43	2456898.50	92.03	54503.12	26695.51	2.14	0.00	947774912	225568
adeptedi	7	88328.22	4.03	404.12	21914.95	0.58	0.01	93155160	8774
dtmail	1	54919.17	5.33	17716.57	10308.94	5.33	0.00	213843968	40192
acroread	8	31218.02	2.67	17744.57	11682.66	0.33	0.00	331454464	11260
dtwm	1	16252.93	2.53	17716.57	6416.05	2.53	0.00	158662656	12848
dtterm	5	4762.71	1.30	76300.29	3658.93	0.26	0.00	33828352	11604
dtaction	23	1389.72	0.33	0.60	4196.43	0.01	0.55	18653184	539
dtsessio	1	1174.87	0.24	17716.57	4932.97	0.24	0.00	23535616	5421
dtcm	1	866.30	0.18	17716.57	4826.21	0.18	0.00	3012096	6490

次の表は、日次コマンド要約のデータを説明したものです。

表 20-5 日次コマンド要約レポート

列	説明
COMMAND NAME	コマンド名。プロセスアカウンティングシステムではオブジェクトモジュールしか報告されないため、シェルプロセスはすべて sh という名前でも取り扱われる。a.out または core と呼ばれるプログラム、またはその他の、適切とは思われない名前のプログラムの使用頻度を監視すると良い。acctcom プログラムを使用して、名前に疑問があるコマンドを誰が実行したか、スーパーユーザー特権が使用されたかどうかを知ることができる
NUMBER CMDS	プライムタイム時間帯に、このコマンドが呼び出された回数
TOTAL KCOREMIN	実行時の毎分あたりにプロセスが使用した K バイトメモリーセグメント数という計量値の累積合計
TOTAL CPU-MIN	このプログラムのプライムタイム時間帯の累積合計処理時間
TOTAL REAL-MIN	このプログラムのプライムタイム時間帯の累積合計実時間 (壁掛け時計)。分単位

表 20-5 日次コマンド要約レポート (続き)

列	説明
MEAN SIZE-K	NUMBER CMDS で表される呼び出し回数に対する TOTAL KCOREMIN の平均
MEAN CPU-MIN	NUMBER CMDS に対する TOTAL CPU-MIN の平均
HOG FACTOR	合計 CPU 時間を経過時間で割った値。システム利用可能時間とシステム使用時間との比であり、プロセスがその実行中に消費する合計利用可能 CPU 時間の相対値を示す
CHARS TRNSFD	読み取りおよび書き込みシステムコールによってプッシュされた文字の合計数。オーバフローのために負の値になることがある
BLOCKS READ	プロセスが実行した物理ブロックの読み取りおよび書き込みの合計数

月次コマンド要約

日次コマンド要約と月次コマンド要約のレポート形式は、実際は同じものです。ただし、日次コマンド要約は現在のアカウント期間だけでレポートするのに対し、月次コマンド要約は会計期間の当初から現在の日付までをレポートします。つまり、月次レポートは、monacct プログラムが最後に実行されたときからの累積データの累積要約を表します。

レポートの例を次に示します。

Oct 16 02:30 2001 MONTHLY TOTAL COMMAND SUMMARY Page 1

COMMAND NAME	NUMBER CMDS	TOTAL COMMAND SUMMARY							
		TOTAL KCOREMIN	TOTAL CPU-MIN	TOTAL REAL-MIN	MEAN SIZE-K	MEAN CPU-MIN	HOG FACTOR	CHARS TRNSFD	BLOCKS READ
TOTALS	42718	4398793.50	361.92	956039.00	12154.09	0.01	0.00	16100942848	825171
netscape	789	3110437.25	121.03	79101.12	25699.58	0.15	0.00	3930527232	302486
adeptedi	84	1214419.00	50.20	4174.65	24193.62	0.60	0.01	890216640	107237
acroread	145	165297.78	7.01	18180.74	23566.84	0.05	0.00	1900504064	26053
dtmail	2	64208.90	6.35	20557.14	10112.43	3.17	0.00	250445824	43280
dtaction	800	47602.28	11.26	15.37	4226.93	0.01	0.73	640057536	8095
soffice.	13	35506.79	0.97	9.23	36510.84	0.07	0.11	134754320	5712
dtwm	2	20350.98	3.17	20557.14	6419.87	1.59	0.00	190636032	14049

月次コマンド要約で提供されるデータの説明については、295 ページの「日次コマンド要約」を参照してください。

最終ログインレポート

このレポートは、特定のログインが最後に使用された日付を示します。この情報を使用して、使用されていないログインやログインディレクトリを見つけることができます。それらのログインやログインディレクトリは保存して削除できます。次に例を示します。

```
Oct 16 02:30 2001 LAST LOGIN Page 1
```

```
01-06-12 kryten          01-09-08 protoA          01-10-14 ripleY
01-07-14 lister          01-09-08 protoB          01-10-15 scutter1
01-08-16 pmorph           01-10-12 rimmer          01-10-16 scutter2
```

acctcom による pacct ファイルの確認

`/var/adm/pacctn` ファイル、または `acct.h` 形式の任意のファイルの内容は、`acctcom` プログラムを使用していつでも調べることができます。このコマンドを実行するときに、ファイルも標準入力も指定しなければ、`acctcom` コマンドは `pacct` ファイルを読み取ります。`acctcom` コマンドで読み取られる各レコードは、終了したプロセスの情報を表します。アクティブなプロセスは、`ps` コマンドを実行して調べます。`acctcom` コマンドのデフォルト出力は次に示す情報を示します。

`acctcom` 出力ファイルを、次に示します。

```
# acctcom
COMMAND
NAME      USER      TTYNAME      START      END          REAL        CPU        MEAN
TIME      TIME      (SECS)      (SECS)      SIZE(K)
#accton   root      ?            02:30:01  02:30:01    0.03        0.01      304.00
turnacct  adm       ?            02:30:01  02:30:01    0.42        0.01      320.00
mv         adm       ?            02:30:01  02:30:01    0.07        0.01      504.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.03        0.01      712.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.01        0.01      824.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.01        0.01      912.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.01        0.01      920.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.01        0.01     1136.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.01        0.01      576.00
closewtm  adm       ?            02:30:01  02:30:01    0.10        0.01      664.00
```

- コマンド名 (# 記号はスーパーユーザー特権を使用して実行された場合)
- ユーザ名
- tty 名 (不明の場合は ? として表示)
- コマンド開始時刻
- コマンド終了時刻
- 実時間 (秒単位)
- CPU 時間 (秒単位)
- 平均サイズ (K バイト単位)

`acctcom` オプションを使用すると、次の情報を得ることができます。

- `fork/exec` フラグの状態 (`exec` を使用しない `fork` の場合は 1)

- システム終了状態
- hog 係数
- 合計 kcore 分
- CPU 係数
- 転送文字数
- 読み取りブロック数

表 20-6 で acctcom のオプションを説明します。

表 20-6 acctcom のオプション

オプション	説明
-a	選択したプロセスに関する特定の平均統計を表示する。統計は出力が記録された後に表示される
-b	ファイルを逆読みし、最後のコマンドから先に表示する。標準入力の読み込みには関係しない
-f	fork/exec フラグおよびシステム終了状態カラムを出力する。出力は 8 進数
-h	平均メモリーサイズの代わりに hog 係数を表示する。これは経過時間とプロセスが実行中に消費した合計 CPU 利用可能時間との比。hog 係数 = 合計 CPU 利用時間/経過時間
-i	入出力カウントを含むカラムを出力する
-k	メモリーサイズの代わりに、キロバイト/分ごとのコアサイズの合計値を表示する
-m	平均コアサイズを表示する。これがデフォルト
-q	平均統計だけを出力する。出力レコードは出力しない
-r	CPU 係数 (システム使用時間 / (システム使用時間 + ユーザー使用時間)) を表示する
-t	システムおよびユーザー CPU 時間を表示する
-v	出力からカラム見出しを除外する
-C sec	合計 (システム + ユーザー) CPU 時間が sec 秒を超えたプロセスだけを表示する
-e time	time 以前に存在したプロセスを hr[:min[:sec]] の書式で表示する
-E time	time 以前に開始されたプロセスを hr[:min[:sec]] の書式で表示する。同じ time を -s と -E の両方に使用すれば、そのときに存在していたプロセスを表示する
-g group	group に属しているプロセスだけを表示する

表 20-6 acctcom のオプション (続き)

オプション	説明
-H <i>factor</i>	<i>factor</i> を超えるプロセスだけを表示する。ただし、 <i>factor</i> は「hog 係数」(-h オプションを参照)
-I <i>chars</i>	<i>chars</i> によって指定されるカットオフ数を超える文字数を転送したプロセスだけを表示する
-l <i>line</i>	端末 /dev/ <i>line</i> に属しているプロセスだけを表示する
-n <i>pattern</i>	<i>pattern</i> 「+」が 1 回以上現れることを意味する以外は、一般的な正規表現に一致するコマンドだけを表示する
-o <i>ofile</i>	レコードを出力しないで、レコードを acct.h 形式で <i>ofile</i> にコピーする
-O <i>sec</i>	CPU システム時間が <i>sec</i> 秒を超えるプロセスだけを表示する
-s <i>time</i>	<i>time</i> 以後に存在したプロセスを hr[:min[: sec]] の書式で表示する
-S <i>time</i>	<i>time</i> 以後に開始されたプロセスを hr[:min[: sec]] の書式で表示する
-u <i>user</i>	<i>user</i> に属しているプロセスだけを表示する

システムアカウントングファイル

/var/adm ディレクトリには、使用中のデータ収集ファイルが含まれています。次の表は、このディレクトリにあるアカウントングファイルを説明しています。

表 20-7 /var/adm ディレクトリ内のファイル

ファイル	説明
dtmp	acctdusg プログラムからの出力
fee	chargefee プログラムからの出力である ASCII の tacct レコード
pacct	現在使用中のプロセスアカウントングファイル
pacctn	turnacct スクリプトの実行によって切り替えられたプロセスアカウントングファイル
Spacctn.MMDD	runacct スクリプトの実行中に生成された MMDD 日付のプロセスアカウントングファイル

/var/adm/acct ディレクトリには、nite、sum、fiscal の各ディレクトリが設けられ、それぞれに実際のデータ収集ファイルが格納されます。たとえば、nite ディレクトリは runacct スクリプトが毎日繰り返して使用するファイルを格納しています。表 20-8 で、/var/adm/acct/nite ディレクトリ内の各ファイルを簡単に説明します。

表 20-8 /var/adm/acct/nite ディレクトリ内のファイル

ファイル	説明
active	runacct スクリプトが進捗状況の記録用、警告メッセージ、エラーメッセージの出力用として使用する
active MMDD	runacct スクリプトがエラーを検出した後は、active ファイルと同じ
cms	prdaily が使用する ASCII の合計コマンド要約
ctacct. MMDD	tacct.h 形式の接続アカウンティングレコード
ctmp	acctcon1 プログラムの出力。ctmp.h 形式の接続セッションレコード (acctcon1 と acctcon2 は互換性を保証するために用意)
daycms	prdaily スクリプトが使用する ASCII 日次コマンド要約
daytacct	tacct.h 形式の 1 日分の合計アカウンティングレコード
disktacct	tacct.h 形式のディスクアカウンティングレコード。dodisk スクリプトが作成する
fd2log	runacct スクリプトの実行中の診断出力
lastdate	runacct スクリプトが最後に実行された日 (date +%m%d 書式)
lock	runacct スクリプトの逐次使用の制御に使用する
lineuse	prdaily スクリプトが使用する tty 回線利用状況レポート
log	acctcon プログラムからの診断出力
log MMDD	runacct スクリプトがエラーを検出した後は、log ファイルと同じ
owtmpx	前日の wtmpx ファイル
reboots	wtmpx ファイルからの開始および終了日付とリブートのリスト
statefile	runacct スクリプトの実行中の現在状態の記録用に使用
tmpwtmp	wtmpfix プログラムが修復した wtmpx ファイル
wtmperror	wtmpfix エラーメッセージが格納される場所
wtmperror MMDD	runacct スクリプトがエラーを検出した後は、wtmperror ファイルと同じ
wtmpx. MMDD	runacct スクリプトが使用する wtmpx ファイルのコピー

sum ディレクトリは、runacct スクリプトが更新し、monacct スクリプトが使用する累積要約ファイルを格納します。次の表で、 /var/adm/acct/sum ディレクトリの中にあるファイルを説明します。

表 20-9 /var/adm/acct/sum ディレクトリ内のファイル

ファイル	説明
cms	バイナリ形式の、会計期の合計コマンド要約ファイル
cmsprev	最新の更新がなされていないコマンド要約ファイル
daycms	内部要約書式の、当日の利用状況を表すコマンド要約ファイル
loginlog	各ユーザーが最後にログインした日付の記録。lastlogin スクリプトによって作成され、prdaily スクリプトの中で使用される
rprrt MMDD	prdaily スクリプトが保存し出力
tacct	会計期の累積合計アカウントティングファイル
tacctprev	最新の更新がない点を除いて tacct と同じ
tacct MMDD	MMDD 日付分の合計アカウントティングファイル

fiscal ディレクトリは monacct スクリプトが作成する定期的要約ファイルを格納します。次の表で、 /var/adm/acct/fiscal ディレクトリの中にあるファイルを説明します。

表 20-10 /var/adm/acct/fiscal ディレクトリ内のファイル

ファイル	説明
cms <i>n</i>	内部要約書式の、会計期 <i>n</i> の合計コマンド要約ファイル
fiscrpt <i>n</i>	会計期 <i>n</i> の rprrt <i>n</i> と同じレポート
tacct <i>n</i>	会計期 <i>n</i> の合計アカウントティングファイル

runacct スクリプトが生成するファイル

次の表で、 /var/adm/acct ディレクトリ内の runacct スクリプトによって生成された最も役に立つファイルを説明します。

表 20-11 runacct が生成するファイル

ファイル	説明
nite/lineuse	runacct スクリプトは acctcon プログラムを呼び出し、 /var/adm/acct/nite/tmpwtmp ファイルから端末の回線利用状況に関するデータを収集し、そのデータを /var/adm/acct/nite/lineuse ファイルに書き込む。 prdaily スクリプトはこれらのデータを使用して回線利用状況を報告する。このレポートは特に不良回線の検出に有効となる。ログアウトとログインの回数の比率が3対1を上回る場合は、回線に障害が起こっている可能性が高い。
nite/daytacct	tacct.h 形式の当日の合計課金ファイル
sum/tacct	毎日の nite/daytacct データの累積が含まれており、課金の目的で使用される。monacct スクリプトは、各月または会計期ごとにデータの累積を新たに開始する。
sum/daycms	runacct スクリプトは acctcms プログラムを呼び出し、当日中に使用されたコマンドを処理し、日次コマンド要約レポートを作成して、/var/adm/acct/sum/daycms ファイルにデータを保存する。ASCII バージョンは /var/adm/acct/nite/daycms ファイル
sum/cms	毎日のコマンド要約の累積。 monacct スクリプトの実行によって新たに累積を開始する。ASCII バージョンは nite/cms ファイル
sum/loginlog	runacct スクリプトは lastlogin スクリプトを呼び出し、 /var/adm/acct/sum/loginlog ファイルのログインの中で最新のログイン日付を更新する。lastlogin コマンドは、すでに有効ではないログインをこのファイルから削除する
sum/rprt.MMDD	runacct スクリプトが実行されるたびに、prdaily スクリプトによって印刷された日次レポートのコピーが保存される

第 21 章

システムパフォーマンスの管理 (トピック)

以下の各章で、システムパフォーマンスの管理について説明します。

第 22 章	システムパフォーマンスについての概要を説明します。
第 23 章	プロセスコマンドを使用してシステムパフォーマンスを向上する手順を説明します。
第 24 章	vmstat、sar、およびディスク利用状態をチェックするコマンドを使用して、システムパフォーマンスを監視する手順を説明します。

第 22 章

システムパフォーマンスの管理 (概要)

コンピュータやネットワークのパフォーマンスを十分に引き出すことは、システム管理における重要な作業です。この章では、コンピュータシステムのパフォーマンスの維持と管理に影響する要素について簡単に説明します。

この章の内容は次のとおりです。

- 308 ページの「システムパフォーマンスに関する情報の参照箇所」
- 309 ページの「システムパフォーマンスおよびシステム資源」
- 309 ページの「プロセスとシステムのパフォーマンス」
- 311 ページの「パフォーマンスの監視」

システムパフォーマンスの管理に関する新機能

ここでは、システムパフォーマンスの管理に使用する新機能について説明します。

DNLC 機能の向上

ディレクトリ名検索キャッシュ (DNLC) が拡張され、1000 以上のファイルを含む大容量のディレクトリにあるファイルにアクセスする際のパフォーマンスが向上しました。

DNLC は一般的なファイルシステムサービスであり、最近参照されたディレクトリ名とそれに関連する v ノードをキャッシュします。UFS ディレクトリのエントリは、ディスク上に直線的に保存されています。よって、エントリを格納するには、その名

前で各エントリを検索する必要があります。新しいエントリを追加する際は、ディレクトリ全体を検索して、その名前が存在していないことを確認する必要があります。このパフォーマンスに関する問題を解決するために、DNLC を使ってディレクトリ全体をメモリーにキャッシュします。

このリリースにおける DNLC のもう 1 つの新機能は、検索したが存在しなかったファイルオブジェクトをキャッシュすることです。これは「ネガティブキャッシュ」と呼びます。ファイルが存在するかどうかを繰り返し調べるアプリケーションに便利です。

詳細は、『Solaris カーネルのチューンアップ・リファレンスマニュアル』を参照してください。

pargs コマンドと preap コマンド

新しい 2 つのコマンド `pargs` と `preap` により、プロセスのデバッグが改善されます。`pargs` コマンドを使用すると、動作中のプロセスまたはコアファイルに関連付けられた引数と環境変数を表示できます。`preap` コマンドを使用すると、終了した (ゾンビ) プロセスを削除できます。

また、`pargs` コマンドを使用することによって、プロセスに渡された引数を `ps` コマンドで一部しか表示できないという問題がようやく解決されました。次のように、`pargs` コマンドを `pgrep` コマンドと併用して、プロセスに渡された引数を表示できます。

```
# pargs `pgrep ttymon`
579: /usr/lib/saf/ttymon -g -h -p system-name console login:
-T sun -d /dev/console -l
argv[0]: /usr/lib/saf/ttymon
argv[1]: -g
argv[2]: -h
argv[3]: -p
argv[4]: system-name console login:
argv[5]: -T
argv[6]: sun
argv[7]: -d
argv[8]: /dev/console
argv[9]: -l
argv[10]: console
argv[11]: -m
argv[12]: ldterm,ttcompat
548: /usr/lib/saf/ttymon
argv[0]: /usr/lib/saf/ttymon
```

`pargs -e` コマンドを使用して、プロセスに関連付けられた環境変数を次の例のように表示できます。

```
$ pargs -e 6763
6763: tcsh
envp[0]: DISPLAY=:0.0
```

pargs コマンドと preap コマンドを使用して、ユーザーの検査権限に含まれるすべてのプロセスを検査できます。スーパーユーザーは、すべてのプロセスを検査できます。

preap コマンドを使用して、終了したプロセスを削除できます。終了したプロセスは、ゾンビプロセスとも呼ばれます。ゾンビプロセスとは、その終了状態がまだ親によって取得または要求されていないプロセスをいいます。これらのプロセスは一般的に支障はありませんが、プロセスの数が多い場合には、システム資源を消費することになります。

preap コマンドの使用方法については、preap(1) のマニュアルページを参照してください。pargs コマンドの使用方法については、proc(1) のマニュアルページを参照してください。

パフォーマンスツール

Solaris 管理コンソールのパフォーマンスツールを使って、システムパフォーマンスおよびシステム資源の使用を監視できます。

Solaris 資源管理によるシステム資源の割り当て、監視、および制御については、『Solaris のシステム管理 (資源管理とネットワークサービス)』の「Solaris 9 リソースマネージャの紹介」を参照してください。

Solaris 管理コンソールの使用と起動の方法については、『Solaris のシステム管理 (基本編)』の「Solaris Management Console の操作 (手順)」を参照してください。

新しい FSS スケジューラと FX スケジューラ

FSS スケジューラ (フェアシェアスケジューラ) および FX スケジューラ (固定優先順位スケジューラ) は、新しいユーザープロセスのスケジューラです。両方のスケジューラでは、タイムシェアリング (TS) スケジューリングクラスおよび対話 (IA) スケジューリングクラスと同じ範囲の優先順位 (0 から 59) を使用します。

FX は、プロセスを確実な優先順位で実行する、固定優先順位スケジューラです。FX はデフォルトのスケジューラではありません。TS では負荷を分散します。FX では負荷を分散しません。同じシステムで FX と TS を使用できますが、FX の優先度を高く設定しすぎると、TS で実行するプロセスが停止状態になります。

FSS を使用すると、CPU 資源の配分を明示的にプロジェクトに割り当てることによって、アプリケーションのパフォーマンスが保証されます。その他の FSS の利点は、次のとおりです。

- システムでの CPU 資源の細かい配分を可能にする。
- 作業負荷に対して重要度に基づいて有効な CPU 資源を割り当てるように制御する。作業負荷の重要度は、各作業負荷に割り当てる CPU 資源の配分数によって表されます。

プロセッサセットを使用するのでなければ、同じシステムで FSS を TS または FX と併用しないでください。プロセッサセットを使用する場合、同じ CPU で競合しないように、各プロセッサセットで実行するすべてのプロセスが同じスケジューリングクラスにある限りは、同じシステムで FSS を TS、IA、FX と併用できます。特に、プロセッサセットを使用して FX クラスのアプリケーションで FSS クラスのアプリケーションの停止状態を招く優先順位を使用しないようにするのでなければ、FX スケジューラは新しい FSS スケジューリングクラスと併用しないでください。

次の表に、上記のスケジューラや、プロジェクト、タスクなどの他の Solaris 資源管理に関する機能、システム資源をより良く管理するための情報を示します。

トピック	参照箇所
FSS スケジューラおよび FX スケジューラの使用法	『Solaris のシステム管理 (資源管理とネットワークサービス)』の「フェアシェアスケジューラ」
FSS	FSS (7) のマニュアルページ
FX	priocntl (1) と dispadmin (1M) のマニュアルページ
プロジェクト	『Solaris のシステム管理 (資源管理とネットワークサービス)』の「プロジェクトとタスク」

システムパフォーマンスに関する情報の参照箇所

作業	参照箇所
プロセスの管理	第 23 章
システムのパフォーマンスの監視	第 24 章
Solaris 調整可能パラメータの変更	『Solaris カーネルのチューンアップ・リファレンスマニュアル』

システムパフォーマンスおよびシステム資源

コンピュータシステムのパフォーマンスは、システムがその資源をどのように使用して割り当てるかによって左右されます。したがって、通常の条件下でどのように動作するかを知るために、システムパフォーマンスを定期的に監視する必要があります。期待できるパフォーマンスについてよく把握し、問題が発生したときに分析できなければなりません。

パフォーマンスに影響を及ぼすシステム資源は次のとおりです。

システム資源	説明
中央処理装置 (CPU)	CPU は、命令をメモリーからフェッチして実行します。
入出力 (I/O) デバイス	I/O デバイスは、コンピュータとの間で情報をやりとりします。この種のデバイスには、端末とキーボード、ディスクドライブ、プリンタなどがあります。
メモリー	物理 (またはメイン) メモリーは、システム上のメモリー (RAM) の容量を示します。

コンピュータシステムの動作とパフォーマンスに関する統計情報を表示するツールについては、第 24 章を参照してください。

プロセスとシステムのパフォーマンス

次の表は、プロセスに関連する用語を示します。

表 22-1 プロセスに関連する用語

用語	説明
プロセス	システムの動作またはジョブ。システムをブートしてコマンドを実行するか、アプリケーションを起動するたびに、システムは 1 つ以上のプロセスをアクティブにする

表 22-1 プロセスに関連する用語 (続き)

用語	説明
軽量プロセス (LWP)	仮想 CPU または実行資源。LWP は、利用できる CPU 資源をスケジューリングクラスと優先順位に基づいて使用するよう、カーネルによってスケジューリングされる。LWP には、カーネルスレッドと LWP が含まれる。カーネルスレッドには、メモリーに常駐する情報が入っている。また、LWP には、スワップ可能な情報が入っている
アプリケーションスレッド	ユーザーのアドレス空間内で独立して実行できる別個のスタックを持った一連の命令。アプリケーションスレッドは LWP の最上部で多重化できる

1 つのプロセスは、複数の軽量プロセスと複数のアプリケーションスレッドで構成できます。カーネルはカーネルスレッド構造をスケジューリングします。この構造は、SunOS 環境内をスケジューリングする実体です。表 22-2 に各種プロセス構造体を示します。

表 22-2 プロセス構造体

構造体	説明
proc	プロセス全体に関連し、メインメモリーに常駐しなければならない情報が入っている
kthread	1 つの LWP に関連し、メインメモリーに常駐しなければならない情報が入っている
user	スワップ可能な、プロセス単位の情報が入っている
klwp	スワップ可能な、LWP プロセス単位の情報が入っている

次の図に、これらのプロセス構造体の関係を示します。

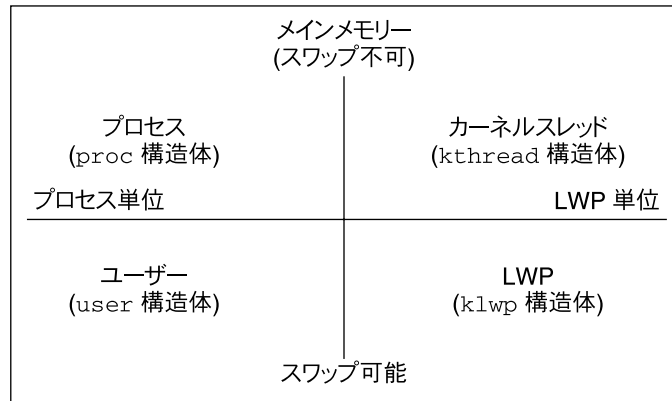


図 22-1 プロセス構造体

プロセス内のすべてのスレッドは、ほとんどのプロセス資源にアクセスできます。ほとんどすべてのプロセスの仮想メモリーが共有されます。あるスレッドが共有データを変更すると、その変更結果をプロセス内の他のスレッドが利用できます。

パフォーマンスの監視

コンピュータの稼働中は、各種のシステム動作を追跡するためにオペレーティングシステムのカウンタが増分されます。追跡されるシステム動作は次のとおりです。

- 中央処理装置 (CPU) の使用状況
- バッファの使用状況
- ディスクとテープの入出力 (I/O) 動作
- 端末デバイスの動作
- システムコールの動作
- コンテキスト切替え
- ファイルアクセス
- 待ち行列の動作
- カーネルテーブル
- プロセス間通信
- ページング
- 空きメモリーとスワップ空間
- カーネルメモリー割り当て (KMA)

監視ツール

Solaris ソフトウェアには、システムパフォーマンスを追跡できるように複数のツールが提供されています。次のような監視ツールがあります。

表 22-3 パフォーマンス監視ツール

コマンド	説明	参照箇所
sar コマンドと sadc コマンド	システム動作データを収集および報告する	第 24 章
ps コマンドと prstat コマンド	活動中のコマンドについての情報を表示する	第 23 章
vmstat コマンドと iostat コマンド	システム動作データの要約。仮想メモリーの統計、ディスクの使用率、CPU の動作など	第 24 章
swap コマンド	ユーザーのシステムで利用可能なスワップ領域についての情報を表示する	『Solaris のシステム管理 (基本編)』の「追加スワップ空間の構成 (手順)」
netstat コマンドと nfsstat コマンド	ネットワークパフォーマンスについての情報を表示する	netstat (1M) と nfsstat (1M) のマニュアルページ
Sun Enterprise SyMON	Sun Enterprise レベルのシステム上で、システム動作データを収集する	『Sun Enterprise SyMON 2.0.1 Software User's Guide』

第 23 章

システムプロセスの管理 (手順)

この章では、システムプロセスを管理する手順について説明します。この章で説明する手順は次のとおりです。

- 315 ページの「プロセスを表示する方法」
- 318 ページの「プロセスに関する情報を表示する方法」
- 319 ページの「プロセスを制御する方法」
- 320 ページの「プロセスを終了させる方法」
- 321 ページの「プロセスクラスに関する基本情報を表示する方法 (priocntl)」
- 322 ページの「プロセスのグローバル優先順位を表示する方法 (ps)」
- 323 ページの「プロセスの優先順位を指定する方法 (priocntl)」
- 323 ページの「タイムシェアリングプロセスのスケジューリングパラメータを変更する方法 (priocntl)」
- 324 ページの「プロセスのクラスを変更する方法 (priocntl)」
- 326 ページの「プロセスの優先順位を変更する方法 (nice)」

システムプロセスの管理に使用するコマンド (概要)

次の表では、プロセス情報を管理するために使用されるコマンドについて説明します。

表 23-1 プロセスを管理するためのコマンド

コマンド	マニュアルページ	説明
ps、 pgrep、 prstat	ps(1)、 pgrep(1)、 prstat(1M)	システム上のアクティブなプロセスの状態をチェックする。また、プロセスについての詳細な情報を表示する

表 23-1 プロセスを管理するためのコマンド (続き)

コマンド	マニュアルページ	説明
dispadmin	dispadmin(1M)	デフォルトのスケジューリングポリシーをリストする
priocntl	priocntl(1)	プロセスに優先順位クラスを割り当て、プロセスの優先度を管理する
nice	nice(1)	タイムシェアリングプロセスの優先度を変更する
psrset	psrset(1M)	特定のプロセスグループを、1つのプロセッサではなく、プロセッサのグループに結合する

Solaris 管理コンソールのプロセスツールを使用すると、ユーザーフレンドリなインタフェースによるプロセスを管理できます。Solaris 管理コンソールの使用と起動の方法については、『Solaris のシステム管理 (基本編)』の「Solaris Management Console の操作 (手順)」を参照してください。

ps コマンド

ps コマンドを使用すると、システム上で活動中のプロセスの状態をチェックできます。また、プロセスについての技術的な情報も表示できます。このデータは、プロセスの優先順位をどのように設定するか判断するなどの管理作業に利用できます。

ps コマンドを使用すると、システム上で活動中のプロセスの状態をチェックできます。使用するオプションに応じて、次の情報が表示されます。

- プロセスの現在の状態
- プロセス ID
- 親プロセス ID
- ユーザー ID
- スケジューリングクラス
- 優先順位
- プロセスのアドレス
- 使用したメモリー
- 使用した CPU 時間

次の表では、ps コマンドを使用して報告されるいくつかのフィールドについて説明します。どのフィールドが表示されるかは、選択するオプションによって異なります。使用可能なすべてのオプションについては、ps(1) のマニュアルページを参照してください。

表 23-2 ps により出力されるフィールド

フィールド	説明
UID	プロセス所有者の実効ユーザー ID
PID	プロセス ID
PPID	親プロセス ID
C	スケジューリングのためのプロセッサ使用率。このフィールドは <code>-c</code> オプションを使用すると表示されない
CLS	プロセスが所属するスケジューリングクラス。リアルタイム、システム、またはタイムシェアリングのいずれか。このフィールドは、 <code>-c</code> オプションを指定した場合にのみ表示される
PRI	カーネルスレッドのスケジューリング優先順位。番号が大きいほど優先順位が高い
NI	プロセスの nice 番号。これは、スケジュール優先順位に影響する。プロセスの nice 番号を大きくすると、その優先順位が下がる
ADDR	proc 構造体のアドレス
SZ	プロセスの仮想アドレスサイズ
WCHAN	プロセスが休眠中のイベントまたはロックのアドレス
STIME	プロセスの起動時刻 (時、分、秒)
TTY	プロセス (またはその親プロセス) が起動された端末。疑問符は、制御端末がないことを示す
TIME	プロセスの起動以降に使用した合計 CPU 時間
CMD	プロセスを生成したコマンド

▼ プロセスを表示する方法

ps コマンドを使用して、システム上で実行中のすべてのプロセスを表示することができます。

```
$ ps [-efc]
```

ps	ログインセッションに関連するプロセスのみを表示する
-ef	システム上で実行中のすべてのプロセスに関する詳細情報を表示する
-c	プロセススケジューラ情報を表示する

例 — プロセスを表示する

次の例は、オプションを指定しないときの `ps` コマンドからの出力を示します。

```
$ ps
  PID TTY          TIME CMD
 1664 pts/4        0:06 csh
 2081 pts/4        0:00 ps
```

次の例は、`ps -ef` の出力を示します。この出力例は、システムのブート時に最初に実行されたプロセスが `sched` (スワップ) であり、それに続いて `init` プロセス、`pageout` の順に実行されたことを示しています。

```
$ ps -ef
  UID  PID  PPID  C   STIME TTY          TIME CMD
  root    0    0    0   Dec 20 ?           0:17 sched
  root    1    0    0   Dec 20 ?           0:00 /etc/init -
  root    2    0    0   Dec 20 ?           0:00 pageout
  root    3    0    0   Dec 20 ?           4:20 fsflush
  root   374   367    0   Dec 20 ?           0:00 /usr/lib/saf/ttymon
  root   367    1    0   Dec 20 ?           0:00 /usr/lib/saf/sac -t 300
  root   126    1    0   Dec 20 ?           0:00 /usr/sbin/rpcbind
  root    54    1    0   Dec 20 ?           0:00 /usr/lib/sysevent/syseventd
  root    59    1    0   Dec 20 ?           0:00 /usr/lib/picl/picld
  root   178    1    0   Dec 20 ?           0:03 /usr/lib/autofs/automountd
  root   129    1    0   Dec 20 ?           0:00 /usr/sbin/keyserv
  root   213    1    0   Dec 20 ?           0:00 /usr/lib/lpsched
  root   154    1    0   Dec 20 ?           0:00 /usr/sbin/inetd -s
  root   139    1    0   Dec 20 ?           0:00 /usr/lib/netsvc/yp/ypbind ...
  root   191    1    0   Dec 20 ?           0:00 /usr/sbin/syslogd
  root   208    1    0   Dec 20 ?           0:02 /usr/sbin/nscd
  root   193    1    0   Dec 20 ?           0:00 /usr/sbin/cron
  root   174    1    0   Dec 20 ?           0:00 /usr/lib/nfs/lockd
daemon  175    1    0   Dec 20 ?           0:00 /usr/lib/nfs/statd
  root   376    1    0   Dec 20 ?           0:00 /usr/lib/ssh/sshd
  root   226    1    0   Dec 20 ?           0:00 /usr/lib/power/powerd
  root   315    1    0   Dec 20 ?           0:00 /usr/lib/nfs/mountd
  root   237    1    0   Dec 20 ?           0:00 /usr/lib/utmpd
  .
  .
  .
```

/proc ファイルシステムとコマンド

プロセスコマンドを使用すると、`/proc` ディレクトリにあるプロセスに関する詳細情報を表示できます。`/proc` ディレクトリはプロセスファイルシステム (PROCFS) とも呼ばれます。アクティブなプロセスのイメージは、そのプロセス ID 番号を使って `/proc` に格納されます。

プロセスツールは `ps` コマンドの一部のオプションに似ていますが、このツールから提供される出力の方が詳細です。プロセスコマンドには次の機能があります。

- `fstat` や `fcntl` 情報、作業ディレクトリ、親プロセスと子プロセスからなるツリーなど、プロセスに関する詳細情報を表示します。
- ユーザーが停止または再開できるように、プロセスに対する制御を提供します。

/proc プロセスコマンドを使用したプロセスの管理

いくつかのプロセスコマンドを使用すると、アクティブなプロセスに関する詳細な技術情報を表示したり、そのプロセスを制御したりできます。表 23-3に、いくつかの `/proc` コマンドを示します。

プロセスが無限ループ内でトラップされた場合や、実行時間が長すぎる場合は、プロセスを終了 (`kill`) できます。 `pkill` コマンドを使用してプロセスを終了する方法については、第 23 章を参照してください。

`/proc` ファイルシステムは、状態情報と制御機能のためのサブディレクトリを含むディレクトリ階層です。

`/proc` ファイルシステムは、ウォッチポイント機能も提供します。この機能は、プロセスのアドレス領域の個々のページの読み取り権または書き込み権を再マップするために使用されます。この機能は制限がなく、MT-safe です。

デバッグ用ツールは、`/proc` の新しいウォッチポイント機能を使用するように変更されています。つまり、ウォッチポイントプロセス全体がより高速になったためです。

`dbx` デバッグ用ツールを使用してウォッチポイントを設定するときの次の制限は取り除かれました。

- SPARC レジスタウィンドウのため、スタック上のローカル変数にウォッチポイントを設定する。
- マルチスレッド化されたプロセスにウォッチポイントを設定する。

詳細については、`proc(4)`、`core(4)`、および `mdb(1)` のマニュアルページを参照してください。

表 23-3 /proc プロセスコマンド

プロセスコマンド	説明
<code>pcred</code>	プロセスの資格情報を表示する
<code>pfiles</code>	プロセス内で開いているファイルに関する <code>fstat</code> 情報と <code>fcntl</code> 情報を表示する
<code>pflags</code>	<code>/proc</code> 追跡フラグ、保留状態のシグナルと保持状態のシグナル、他の状態情報を出力する

表 23-3 /proc プロセスコマンド (続き)

プロセスコマンド	説明
<code>pldd</code>	プロセスにリンクされている動的ライブラリを表示する
<code>pmap</code>	各プロセスのアドレス空間マップを表示する
<code>psig</code>	各プロセスのシグナルの動作とハンドラを表示する
<code>prun</code>	各プロセスを開始する
<code>pstack</code>	各プロセス内の LWP の 16 進とシンボルのスタックトレースを出力する
<code>pstop</code>	各プロセスを停止する
<code>ptime</code>	<code>microstate</code> アカウントを使用してプロセスの時間を測定する
<code>ptree</code>	プロセスを含むプロセスツリーを表示する
<code>pwait</code>	プロセス終了後の状態情報を表示する
<code>pwdx</code>	プロセスの現在の作業ディレクトリを表示する

詳細については、`proc(1)` のマニュアルページを参照してください。

▼ プロセスに関する情報を表示する方法

1. 詳細を表示したいプロセスのプロセス ID を表示します。

```
# pgrep process
```

`process` は、詳細を表示したいプロセスの名前です。

プロセス ID は、出力の第 1 列目に表示されます。

2. 表示したいプロセス情報を表示します。

```
# /usr/bin/pcommand pid
```

`pcommand` 実行したいプロセスツールコマンド。表 23-3 を参照

`pid` プロセス ID

例 — プロセスに関する情報を表示する

次の例は、プロセスツールコマンドを使用して `lpsched` プロセスに関する詳細情報を表示する方法を示しています。

```
# pgrep lpsched 1
213
```

```

# pwdx 213 2
213: /
# ptree 213 3
213 /usr/lib/lpsched
# pfiles 213 4
213: /usr/lib/lpsched
Current rlimit: 4096 file descriptors
0: S_IFIFO mode:0000 dev:270,0 ino:67 uid:0 gid:0 size:0
   O_RDWR
1: S_IFIFO mode:0000 dev:270,0 ino:67 uid:0 gid:0 size:0
   O_RDWR
3: S_IFCHR mode:0666 dev:136,0 ino:35882 uid:0 gid:3 rdev:21,0
   O_WRONLY FD_CLOEXEC
4: S_IFDOOR mode:0444 dev:275,0 ino:18526 uid:0 gid:0 size:0
   O_RDONLY|O_LARGEFILE FD_CLOEXEC door to nscd[208]
5: S_IFREG mode:0664 dev:136,0 ino:64648 uid:71 gid:8 size:0
   O_WRONLY

```

1. lpsched のプロセス ID を表示します。
2. lpsched の現在の作業ディレクトリを表示します。
3. lpsched が入っているプロセスツリーを表示します。
4. fstat と fcntl の情報を表示します。

▼ プロセスを制御する方法

1. 制御するプロセスのプロセス ID を表示します。

```
# pgrep process
```

process は、制御するプロセスの名前です。

プロセスの ID は、出力の第 1 列目に表示されます。

2. 適切なプロセスコマンドを使用してプロセスを制御します。

```
# /usr/bin/pcommand pid
```

pcommand

実行したいプロセスコマンド。表 23-3 を参照

pid

プロセス ID

3. プロセス状態を確認します。

```
# ps -ef | grep PID
```

例 — プロセスを制御する

次の例は、プロセスツールを使用して dtpad を停止し再開する方法を示しています。

```
# pgrep dtpad 1
2921
```

プロセスクラス情報の管理

次のリストは、システム上で構成されるクラスと、タイムシェアリングクラスの利用者優先順位の範囲です。クラスの種類は次のとおりです。

- フェアシェア (FSS)
- 固定優先順位 (FX)
- システム (SYS)
- 対話型 (IA)
- リアルタイム (RT)
- タイムシェアリング (TS)
 - ユーザーが指定する -60 から +60 までの優先順位の範囲
 - プロセスの優先順位は、親プロセスから継承されます。これを「ユーザーモード」の優先順位と呼びます。
 - システムは、ユーザーモードの優先順位をタイムシェアリングディスパッチパラメータテーブル内で検索し、nice または `priocntl` (ユーザー提供) 優先順位に追加し、0 から 59 までの範囲を確保して「グローバル」優先順位を作成します。

プロセスのスケジューリング優先順位を変更する (`priocntl`)

プロセスのスケジューリング優先順位とは、プロセススケジューラによって割り当てられる優先順位のことです。これらの優先順位は、スケジューラのスケジューリングポリシーに従って割り当てられます。`dispadm` コマンドを使用すると、デフォルトのスケジューリングポリシーを表示できます。

`priocntl` コマンドを使用して、プロセスを優先順位クラスに割り当てて、プロセスの優先順位を管理できます。`priocntl` コマンドを使用してプロセスを管理する手順については、323 ページの「プロセスの優先順位を指定する方法 (`priocntl`)」を参照してください。

▼ プロセスクラスに関する基本情報を表示する方法 (`priocntl`)

`priocntl -l` コマンドを使用すると、プロセスのスケジューリングクラスと優先順位の範囲を表示できます。

```
$ priocntl -l
```

例—プロセスクラスに関する基本情報を表示する (priocntl)

次の例に `priocntl -l` コマンドからの出力を示します。

```
# priocntl -l
CONFIGURED CLASSES
=====

SYS (System Class)

TS (Time Sharing)
    Configured TS User Priority Range: -60 through 60

FX (Fixed priority)
    Configured FX User Priority Range: 0 through 60

IA (Interactive)
    Configured IA User Priority Range: -60 through 60
```

▼ プロセスのグローバル優先順位を表示する方法 (ps)

`ps` コマンドを使用して、プロセスのグローバル優先順位を表示できます。

```
$ ps -ecl
```

グローバル優先順位は、PRI カラムの下に表示されます。

例—プロセスのグローバル優先順位を表示する (ps)

次の例は、`ps -ecl` コマンドの出力を示します。PRI カラム内のデータは、`pageout` プロセスが最上位の優先順位を持ち、`sh` が最下位の優先順位であることを示しています。

```
$ ps -ecl
 F S UID PID PPID CLS PRI ADDR SZ WCHAN TTY TIME COMD
19 T 0 0 0 0 SYS 96 f00d05a8 0 ? 0:03 sched
 8 S 0 1 0 TS 50 ff0f4678 185 ff0f4848 ? 36:51 init
19 S 0 2 0 0 SYS 98 ff0f4018 0 f00c645c ? 0:01 pageout
19 S 0 3 0 0 SYS 60 ff0f5998 0 f00d0c68 ? 241:01 fsflush
 8 S 0 269 1 TS 58 ff0f5338 303 ff49837e ? 0:07 sac
 8 S 0 204 1 TS 43 ff2f6008 50 ff2f606e console 0:02 sh
```

▼ プロセスの優先順位を指定する方法 (priosctl)

1. スーパーユーザーになります。
2. 指定した優先順位でプロセスを起動します。

```
# priosctl -e -c class -m userlimit -p pri command-name
```

-e	コマンドを実行する
-c class	プロセスを実行する範囲のクラスを指定する。有効なクラスは TS (タイムシェアリング)、RT (リアルタイム)、IA (対話型)、FSS (フェアシェア)、または FX (固定優先順位) です。
-m userlimit	-p オプションを使用するとき、優先順位を上下できる最大範囲を指定する
-p pri command-name	リアルタイムスレッド用に RT クラス内で相対優先順位を指定できるようにする。タイムシェアリングプロセスの場合は、-p オプションを使用すると -60 から +60 までのユーザー指定の優先順位を指定できる

3. プロセス状態を確認します。

```
# ps -ecl | grep command-name
```

例—プロセスの優先順位を指定する (priosctl)

次の例では、ユーザーが指定する最上位の優先順位を使用して find コマンドを開始します。

```
# priosctl -e -c TS -m 60 -p 60 find . -name core -print
# ps -ecl | grep find
```

▼ タイムシェアリングプロセスのスケジューリングパラメータを変更する方法 (priosctl)

1. スーパーユーザーになります。
2. 実行中のタイムシェアリングプロセスのスケジューリングパラメータを変更します。

```
# priosctl -s -m userlimit [-p userpriority] -i idtype idlist
```

-s	ユーザー優先順位の範囲について上限を設定し、現在の優先順位を変更する
----	------------------------------------

<code>-m userlimit</code>	<code>-p</code> オプションを使用するとき、優先順位を上下できる最大範囲を指定する
<code>-p userpriority</code>	優先順位を指定する
<code>-i idtype idlist</code>	<code>idtype</code> と <code>idlist</code> の組み合わせを使用してプロセスを識別する。 <code>idtype</code> では <code>PID</code> や <code>UID</code> など、ID のタイプを指定する。 <code>idlist</code> を使用して、 <code>pid</code> または <code>UID</code> のリストを識別する

3. プロセス状態を確認します。

```
# ps -ecl | grep idlist
```

例—タイムシェアリングプロセスのスケジューリングパラメータを変更する (`prctl`)

次の例では、500 ミリ秒のタイムスライス、クラス `RT` 内の優先順位 20、グローバル優先順位 120 を指定して、コマンドを実行します。

```
# prctl -e -c RT -t 500 -p 20 myprog
# ps -ecl | grep myprog
```

▼ プロセスのクラスを変更する方法 (`prctl`)

1. (省略可能) スーパーユーザーになります。

注—プロセスをリアルタイムプロセスに変更したり、リアルタイムプロセスから変更したりするには、ユーザーはスーパーユーザーであるか、リアルタイムシェル内で作業中でない限りなりません。

2. プロセスのクラスを変更します。

```
# prctl -s -c class -i idtype idlist
```

<code>-s</code>	ユーザー優先順位の範囲について上限を設定し、現在の優先順位を変更する
<code>-c class</code>	クラス <code>TS</code> またはクラス <code>RT</code> を指定して、プロセスのクラスを変更する
<code>-i idtype idlist</code>	<code>idtype</code> と <code>idlist</code> の組み合わせを使用してプロセスを識別する。 <code>idtype</code> では <code>PID</code> や <code>UID</code> など、ID のタイプを指定する。 <code>idlist</code> を使用して、 <code>pid</code> または <code>UID</code> のリストを識別する

3. プロセス状態を確認します。

```
# ps -ecl | grep idlist
```

例—プロセスのクラスを変更する (priosctl)

次の例では、ユーザー 15249 が所有するすべてのプロセスをリアルタイムプロセスに変更します。

```
# priosctl -s -c RT -i uid 15249
# ps -ecl | grep 15249
```

注—スーパーユーザーとしてユーザープロセスをリアルタイムクラスに変更すると、そのユーザーは `priosctl -s` を使用して、リアルタイムのスケジューリングパラメータを変更できません。

タイムシェアリングプロセスの優先順位を変更する (nice)

`nice` コマンドは、SunOS の旧バージョンとの下位互換性を保つためののみサポートされます。 `priosctl` コマンドを使用する方がプロセスを柔軟に管理できます。

プロセスの優先順位は、そのスケジュールクラスポリシーと `nice` 番号によって決定されます。各タイムシェアリングプロセスは、グローバル優先順位を持っています。グローバル優先順位は、ユーザーが指定した優先順位 (`nice` コマンドまたは `priosctl` コマンドの影響を受ける) とシステムで計算された優先順位を加算して算出されます。

プロセスの実行優先順位番号は、オペレーティングシステムによって割り当てられ、スケジュールクラス、使用される CPU 時間、 `nice` 値 (タイムシェアリングプロセスの場合) などの複数の要素によって決定されます。

各タイムシェアリングプロセスは、親プロセスから継承したデフォルトの `nice` 番号で起動します。 `nice` 値は、 `ps` レポートの `NI` カラムに表示されます。

ユーザーは、自分が与える `nice` 番号優先順位を大きくしてプロセスの優先順位を下げるすることができます。ただし、 `nice` 番号を小さくしてプロセスの優先順位を上げることができるのは、スーパーユーザー (または `root`) だけです。これは、ユーザーが各自のプロセスの優先順位を大きくして CPU の独占比率を高めるのを防ぐためです。

`nice` 番号の範囲は 0 から +39 までで、0 が最上位の優先順位となります。各タイムシェアリングプロセスのデフォルト `nice` 値は 20 です。 `nice` コマンドには利用できるバージョンが 2 つあり、1 つは標準バージョンの `/usr/bin/nice` で、もう 1 つは C シェル組み込みコマンドです。

▼ プロセスの優先順位を変更する方法 (nice)

注 - この節では /usr/bin/nice コマンドの構文についてだけ説明し、C シェル nice 組み込みコマンドについての説明は行いません。C シェル nice コマンドについては、csh(1) のマニュアルページを参照してください。

1. まず、ユーザーとしてコマンドの優先順位を下げるのか、あるいはスーパーユーザーとしてコマンドの優先順位を上げるのか下げるのかを決定してから、次のうちの1つを選択します。
 - 手順2の例に従って、ユーザーとしてコマンドの優先順位を下げる。
 - 手順3に従って、スーパーユーザーとしてコマンドの優先順位を上げるか、または下げる。

2. ユーザーとして、nice 番号を大きくすることでコマンドの優先順位を下げます。次の nice コマンドは、nice 番号を5単位分大きくすることで、*command-name* を実行する優先順位を下げます。

```
$ /usr/bin/nice -5 command-name
```

上記のコマンドでは、マイナス記号は次にくるものがオプションであることを表します。このコマンドは、次のように指定することもできます。

```
% /usr/bin/nice -n 5 command-name
```

次の nice コマンドは、nice 番号をデフォルトの10単位分大きくすることで、*command-name* の優先順位を下げます。ただし、最高値の39を超えさせることはできません。

```
% /usr/bin/nice command-name
```

3. スーパーユーザーとして、nice 番号を変更することでコマンドの優先順位を上げるか、または下げます。

次の nice コマンドは、nice 番号を10単位分小さくすることで、*command-name* の優先順位を上げます。ただし、最低値の0未満にすることはできません。

```
# /usr/bin/nice --10 command-name
```

上記のコマンドでは、最初のマイナス記号は次にくるものがオプションであることを表します。2番目のマイナス記号は負の数を表します。

次の nice コマンドは、nice 番号を5単位分大きくすることで、*command-name* の優先順位を下げます。ただし、最高値の39を超えさせることはできません。

```
# /usr/bin/nice -5 command-name
```

詳細については、nice(1) のマニュアルページを参照してください。

システムのプロセスに関する問題解決方法

すでに判明している問題の解決方法のヒントを次に示します。

- 同じユーザーが所有する複数の同じジョブがないかどうかを調べます。ジョブが終了するまで待たずに多数のバックグラウンドジョブを起動するスクリプトを実行した場合に、この問題が発生することがあります。
- CPU 時間が大量に増えているプロセスがないかどうかを調べます。この問題を調べるには、`ps` 出力の `TIME` フィールドを確認します。そのプロセスが無限ループに入っている可能性があります。
- 実行中のプロセスの優先順位が高すぎないかどうかを調べます。`ps -c` コマンドを使用して `CLS` フィールドを調べると、各プロセスのスケジューラクラスが表示されます。リアルタイム (RT) プロセスとして実行中のプロセスが CPU を独占している可能性があります。また、`nice` 値の大きいタイムシェアリング (TS) プロセスがないかどうかを調べます。スーパーユーザー特権を持つユーザーが、このプロセスの優先順位を上げすぎた可能性があります。システム管理者は、`nice` コマンドを使用して優先順位を下げるすることができます。
- 制御がきかなくなったプロセスを調べます。このようなプロセスは、CPU 時間の使用が継続的に増加していきます。プロセスが開始 (`STIME`) されたときに調べるか、しばらくの間 CPU 時間 (`TIME`) が累計されるのを観察していると、この問題が発生しているかどうかを判断できます。

第 24 章

システムパフォーマンスの監視 (手順)

この章では、`vmstat`、`iostat`、`df`、または `sar` コマンドを使用してシステムパフォーマンスを監視する手順について説明します。この章で説明する手順は次のとおりです。

- 331 ページの「仮想メモリーの統計情報を表示する方法 (`vmstat`)」
- 332 ページの「システムイベント情報を表示する方法 (`vmstat -s`)」
- 333 ページの「スワップの統計情報を表示する方法 (`vmstat -S`)」
- 333 ページの「キャッシュフラッシュの統計情報を表示する方法 (`vmstat -c`)」
- 334 ページの「各デバイス当りの割り込み数を表示する方法 (`vmstat -i`)」
- 335 ページの「ディスクの使用状況を表示する方法 (`iostat`)」
- 336 ページの「拡張ディスク統計情報を表示する方法 (`iostat -xtc`)」
- 337 ページの「ディスク容量情報を表示する方法 (`df`)」
- 338 ページの「ファイルアクセスをチェックする方法 (`sar -a`)」
- 339 ページの「バッファ動作をチェックする方法 (`sar -b`)」
- 340 ページの「システムコールの統計情報をチェックする方法 (`sar -c`)」
- 341 ページの「ディスク動作をチェックする方法 (`sar -d`)」
- 343 ページの「ページアウトとメモリーをチェックする方法 (`sar -g`)」
- 344 ページの「カーネルメモリーの割り当てをチェックする方法 (`sar -k`)」
- 345 ページの「プロセス間通信をチェックする方法 (`sar -m`)」
- 346 ページの「ページイン動作をチェックする方法 (`sar -p`)」
- 347 ページの「待ち行列動作をチェックする方法 (`sar -q`)」
- 348 ページの「未使用のメモリーをチェックする方法 (`sar -r`)」
- 349 ページの「CPU の使用状況をチェックする方法 (`sar -u`)」
- 351 ページの「システムテーブルの状態をチェックする方法 (`sar -v`)」
- 352 ページの「スワップ動作をチェックする方法 (`sar -w`)」
- 353 ページの「端末動作をチェックする方法 (`sar -y`)」
- 354 ページの「システム全体のパフォーマンスをチェックする方法 (`sar -A`)」
- 357 ページの「自動データ収集を設定する方法」

システムパフォーマンスの監視に関する 新機能

vmstat コマンドには、`faults-in` 列によるすべてのシステム割り込みが追加されています。

従来の Solaris リリースでは、この列でクロックデバイスの割り込みが報告されませんでした。

詳細については、`vmstat (1M)` のマニュアルページを参照してください。

仮想メモリーの統計情報の表示 (vmstat)

vmstat コマンドを使用すると、仮想メモリーの統計情報と、CPU の負荷、ページング、コンテキスト切替え数、デバイス割り込み、システムコールなどのシステムイベントに関する情報を表示できます。また、vmstat コマンドを使用すると、スワップ、キャッシュフラッシュ、および割り込みに関する統計情報も表示できます。

次の表に vmstat の出力内のフィールドを示します。

表 24-1 vmstat コマンドからの出力

カテゴリ	フィールド名	説明
procs		次の状態を報告する
	r	ディスパッチ待ち行列内のカーネルスレッド数
	b	資源を待機中のブロックされたカーネルスレッド数
	w	資源処理の完了を待機中のスワップアウトされた軽量プロセス数
memory		実メモリーと仮想メモリーの使用状況を表示する
	swap	使用可能なスワップ空間
	free	空きリストのサイズ
page		ページフォルトとページング動作を 1 秒当りの単位数として表示する

表 24-1 vmstat コマンドからの出力 (続き)

カテゴリ	フィールド名	説明
	re	回収されたページ数
	mf	軽度のフォルトと重大なフォルト
	pi	ページインされたキロバイト数
	po	ページアウトされたキロバイト数
	fr	解放されたキロバイト数
	de	最後にスワップインされたプロセスに必要だと予想されるメモリー
	sr	ページデーモンによって走査され、現在は使用されていないページ数。sr が 0 以外の値であれば、ページデーモンは実行されている
disk		最高 4 台のディスク上のデータを示す、1 秒当りのディスク処理数を表示する
faults		トラップ/割り込み率 (1 秒当り) を表示する
	in	1 秒当りの割り込み数
	sy	1 秒当りのシステムコール数
	cs	CPU のコンテキスト切替え率
cpu		CPU 時間の使用状況を表示する
	us	ユーザー時間
	sy	システム時間
	id	アイドル時間

このコマンドの詳細については、vmstat (1M) のマニュアルページを参照してください。

▼ 仮想メモリーの統計情報を表示する方法 (vmstat)

秒単位で時間間隔を指定して vmstat コマンドを使用すると、仮想メモリーの統計情報が収集されます。

```
$ vmstat n
```

n は、報告間隔を秒単位で表した時間間隔です。

例 — 仮想メモリの統計情報を表示する

次の例に、5 秒間隔で収集された統計情報に関する `vmstat` の表示を示します。

```
$ vmstat 5
procs      memory                page                disk                faults            cpu
r  b  w  swap free re  mf  pi  po  fr de sr f0 s3 -- -- in sy cs us sy id
0  0  8 28312 668 0   9   2   0   1  0  0  0  1  0  0  10 61 82  1  2 97
0  0  3 31940 248 0  10  20   0  26  0 27  0  4  0  0  53 189 191  6  6 88
0  0  3 32080 288 3  19  49   6  26  0 15  0  9  0  0  75 415 277  6 15 79
0  0  3 32080 256 0  26  20   6  21  0 12  1  6  0  0 163 110 138  1  3 96
0  1  3 32060 256 3  45  52  28  61  0 27  5 12  0  0 195 191 223  7 11 82
0  0  3 32056 260 0   1   0   0   0  0  0  0  0  0  0  4  52  84  0  1 99
```

▼ システムイベント情報を表示する方法 (`vmstat -s`)

`vmstat -s` コマンドを実行すると、システムを前回ブートした後に発生した各種システムイベントの合計が表示されます。

```
$ vmstat -s
      0 swap ins
      0 swap outs
      0 pages swapped in
      0 pages swapped out
392182 total address trans. faults taken
20419 page ins
  923 page outs
30072 pages paged in
 9194 pages paged out
65167 total reclaims
65157 reclaims from free list
      0 micro (hat) faults
392182 minor (as) faults
19383 major faults
85775 copy-on-write faults
66637 zero fill page faults
46309 pages examined by the clock daemon
      6 revolutions of the clock hand
15578 pages freed by the clock daemon
4398 forks
  352 vforks
4267 execs
12926285 cpu context switches
109029866 device interrupts
499296 traps
22461261 system calls
778068 total name lookups (cache hits 97%)
 18739 user   cpu
 34662 system cpu
52051435 idle  cpu
 25252 wait  cpu
```

▼ スワップの統計情報を表示する方法 (vmstat -S)

vmstat -S を実行すると、スワップの統計情報が表示されます。

```
$ vmstat -S
procs      memory          page          disk          faults        cpu
r  b  w  swap  free  si  so pi po fr de sr f0 s0 s6 --  in  sy  cs us sy id
0  0  0 200968 17936  0  0  0  0  0  0  0  0  0  0  109  43  24  0  0 100
```

スワッピング統計情報を次の表に示します。その他のフィールドの説明については、表 24-1 を参照してください。

表 24-2 vmstat -S コマンドからの出力

フィールド名	説明
si	1 秒当りにスワップされた平均軽量プロセス数
so	スワップアウトされた全プロセス数

注 - vmstat コマンドは、これらの両フィールドを出力しません。スワップ統計情報の詳細情報を表示するには、sar コマンドを使用してください。

▼ キャッシュフラッシュの統計情報を表示する方法 (vmstat -c)

vmstat -c を実行すると、仮想キャッシュのキャッシュフラッシュ統計情報が表示されます。

```
$ vmstat -c
usr      ctx      rgn      seg      pag      par
0      60714      5 134584 4486560 4718054
```

出力には、今回のブート後に発生したキャッシュフラッシュの合計数が表示されます。キャッシュタイプを次の表に示します。

表 24-3 vmstat -c コマンドからの出力

キャッシュ名	キャッシュタイプ
usr	ユーザー
ctx	コンテキスト
rgn	領域

表 24-3 vmstat -c コマンドからの出力 (続き)

キャッシュ名	キャッシュタイプ
seg	セグメント
pag	ページ
par	ページの一部

▼ 各デバイス当りの割り込み数を表示する方法 (vmstat -i)

vmstat -i コマンドを実行すると、デバイス当りの割り込み数が表示されます。

```
$ vmstat -i
```

例 — 各デバイス当りの割り込み数を表示する

次の例は、vmstat -i コマンドからの出力を示します。

```
$ vmstat -i
interrupt          total          rate
-----
clock              52163269      100
esp0                2600077       4
zsc0                25341         0
zsc1                48917         0
cgsixc0            459           0
lec0               400882         0
fdc0                14            0
bppc0              0              0
audiocs0           0              0
-----
Total              55238959      105
```

ディスク使用状況の表示 (iostat n)

iostat コマンドを使用すると、ディスクの入出力に関する統計情報を表示し、スループット、使用率、待ち行列の長さ、トランザクション率、サービス時間の計測結果を表示できます。このコマンドの詳細は、iostat (1M) のマニュアルページを参照してください。

▼ ディスクの使用状況を表示する方法 (iostat)

秒単位で時間間隔を指定して `iostat` コマンドを使用すると、ディスクの使用状況が表示されます。

```
$ iostat 5
      tty          fd0          sd3          nfs1          nfs31          cpu
tin tout kps tps serv kps tps serv kps tps serv kps tps serv us sy wt id
  0   1   0   0  410    3   0  29    0   0   9    3   0  47   4   2   0  94
```

出力の 1 行目は、今回のブート以降の統計情報を示します。2 行目以降は、時間間隔ごとの統計情報を示します。デフォルトでは、端末 (tty)、ディスク (fd と sd)、CPU (cpu) の統計情報が表示されます。

次の表に `iostat` コマンド出力内のフィールドを示します。

表 24-4 `iostat n` コマンドからの出力

デバイスの種類	フィールド名	説明
端末		
	tin	端末の入力待ち行列内の文字数
	tout	端末の出力待ち行列内の文字数
ディスク		
	bps	1 秒当りのブロック数
	tps	1 秒当りのトランザクション数
	serv	ミリ秒単位で表した平均サービス時間
CPU		
	us	ユーザーモード
	sy	システムモード
	wt	入出力待機中
	id	アイドル状態

例 — ディスクの使用状況を表示する

次の例は、5 秒間隔で収集されるディスク統計情報を示します。

```
$ iostat 5
      tty          sd0          sd6          nfs1          nfs49          cpu
tin tout kps tps serv kps tps serv kps tps serv kps tps serv us sy wt id
  0   0   1   0  49    0   0   0    0   0   0    0   0  15   0   0   0  100
  0  47   0   0   0    0   0   0    0   0   0    0   0   0   0   0   0  100
  0  16   0   0   0    0   0   0    0   0   0    0   0   0   0   0   0  100
  0  16   0   0   0    0   0   0    0   0   0    0   0   0   0   0   0  100
```

```

0 16 44 6 132 0 0 0 0 0 0 0 0 0 0 0 0 1 99
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
0 16 3 1 23 0 0 0 0 0 0 0 0 0 0 0 0 1 99
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100

```

▼ 拡張ディスク統計情報を表示する方法 (iostat -xtc)

iostat -xtc コマンドを実行すると、拡張ディスク統計情報が表示されます。

```

$ iostat -xtc
                extended device statistics
device          r/s      w/s      kr/s      kw/s wait actv  svc_t  %w  %b  tin tout  us sy wt id
fd0             0.0      0.0      0.0      0.0 0.0 0.0   0.0  0  0   0  0  0  0  0 100
sd0             0.0      0.0      0.4      0.4 0.0 0.0  49.5  0  0
sd6             0.0      0.0      0.0      0.0 0.0 0.0   0.0  0  0
nfs1            0.0      0.0      0.0      0.0 0.0 0.0   0.0  0  0
nfs49           0.0      0.0      0.0      0.0 0.0 0.0  15.1  0  0
nfs53           0.0      0.0      0.4      0.0 0.0 0.0  24.5  0  0
nfs54           0.0      0.0      0.0      0.0 0.0 0.0   6.3  0  0
nfs55           0.0      0.0      0.0      0.0 0.0 0.0   4.9  0  0

```

iostat -xtc コマンドを使用すると、ディスクごとに 1 行ずつ出力が表示されます。出力フィールドを次の表に示します。

表 24-5 iostat -xtc コマンドからの出力

フィールド名	説明
r/s	1 秒当りの読み取り数
w/s	1 秒当りの書き込み数
kr/s	1 秒当りの読み取りキロバイト数
kw/s	1 秒当りの書き込みキロバイト数
wait	サービス (待ち行列の長さ) を待機中の平均トランザクション数
actv	サービス中の平均トランザクション数
svc_t	ミリ秒単位で表した平均サービス時間
%w	待ち行列が空でない時間の割合

表 24-5 iostat -xtc コマンドからの出力 (続き)

フィールド名	説明
%b	ディスクがビジーである時間の割合

ディスク容量統計情報の表示 (df)

df コマンドを使用すると、マウントされている各ディスク上の空きディスク容量が表示されます。レポート用の統計情報では使用可能容量の合計の内先頭に 10% の空き容量を残しておくので、df から報告される「使用可能」ディスク容量は全容量の 90% のみに相当します。この先頭の空き容量は、パフォーマンスを高めるために常に空になっています。

実際に df コマンドからレポートされるディスク容量の割合は、使用済み容量を使用可能容量で割った値です。

ファイルシステムの容量が 90% を超える場合は、cp コマンドを使用して空いているディスクにファイルを転送するか、tar または cpio コマンドを使用してテープに転送できます。または、ファイルを削除できます。

このコマンドの詳細については、df (1M) のマニュアルページを参照してください。

▼ ディスク容量情報を表示する方法 (df)

df -k コマンドを使用すると、ディスク容量情報がキロバイト単位で表示されます。

```
$ df -k
Filesystem          kbytes    used   avail capacity  Mounted on
/dev/dsk/c0t3d0s0  192807   40231  133296    24%      /
```

次の表に df -k の出力内のフィールドを示します。

表 24-6 df -k コマンドからの出力

フィールド名	説明
kbytes	ファイルシステム内の使用可能容量の合計
used	使用されている容量
avail	使用可能容量
capacity	使用されている容量が全容量に占める割合

表 24-6 df -k コマンドからの出力 (続き)

フィールド名	説明
mounted on	マウントポイント

例 — ファイルシステム情報を表示する

次の例は、df -k コマンドの出力を示します。

```
$ df -k
Filesystem            kbytes    used    avail capacity  Mounted on
/dev/dsk/c0t0d0s0     384120   131596   214112    39%      /
/dev/dsk/c0t0d0s6    1388419 1050390  282493    79%     /usr
/proc                  0         0         0         0%     /proc
mnttab                 0         0         0         0%     /etc/mnttab
fd                     0         0         0         0%     /dev/fd
swap                   467152     40    467112     1%     /var/run
swap                   467160     48    467112     1%     /tmp
/dev/dsk/c0t0d0s4    1784644 1525360  205745    89%     /export
venus:/usr/dist       20612581 13237316 6963015    66%     /usr/dist
```

システム動作の監視 (sar)

次のような目的には、sar コマンドを使用します。

- システム動作についてのデータを編成し表示する
- 特殊な要求に基づいて、システム動作データにアクセスする
- システムパフォーマンスを測定および監視するレポートを自動的に生成する。また、特定のパフォーマンス障害を正確に突き止めるための、特殊な要求レポートも生成する。これらのツールについては、354 ページの「システム動作データの自動収集 (sar)」を参照

このコマンドの詳細は、sar (1) のマニュアルページを参照してください。

▼ ファイルアクセスをチェックする方法 (sar -a)

sar -a コマンドを使用すると、ファイルアクセス操作の統計情報が表示されます。

```
$ sar -a
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00  iget/s namei/s dirbk/s
01:00:00          0          0          0
```

```

02:00:02      0      0      0
03:00:00      0      1      0
04:00:00      0      0      0
05:00:01      0      0      0
06:00:00      0      0      0

Average      0      1      0

```

次の表に、`sar -a` コマンドによってレポートされるオペレーティングシステムのルーチンを示します。

表 24-7 `sar -a` コマンドからの出力

フィールド名	説明
<code>iget/s</code>	ディレクトリ名検索キャッシュ (DNLC) 内に入っていない <code>i</code> ノードに対して出された要求数
<code>namei/s</code>	1 秒当りのファイルシステムパスの検索数。namei で DNLC 内にディレクトリ名が見つからない場合は、iget が呼び出され、ファイルまたはディレクトリの <code>i</code> ノードが取得される。したがって、ほとんどの <code>igets</code> は DNLC が欠落した結果である
<code>dirbk/s</code>	1 秒間に実行されたディレクトリブロックの読み取り回数

表示される値が大きいほど、カーネルはユーザーファイルへのアクセスに長い時間を費やしています。この時間には、プログラムとアプリケーションによるファイルシステムの使用量が反映されます。`-a` オプションを使用すると、アプリケーションのディスク依存度を表示できるので便利です。

▼ バッファー動作をチェックする方法 (`sar -b`)

`sar -b` コマンドを使用すると、バッファー動作の統計情報が表示されます。

バッファーは、`i` ノード、シリンダグループブロック、間接ブロックなどのメタデータをキャッシュするために使用されます。

```

$ sar -b
00:00:00 bread/s lread/s %rcache bwrit/s lwrit/s %wcache pread/s pwrit/s
01:00:00      0      0      100      0      0      55      0      0

```

次の表は、`-b` オプションを指定したときに表示されるバッファー動作を示します。

表 24-8 `sar -b` コマンドからの出力

フィールド名	説明
<code>bread/s</code>	ディスクからバッファークッシュに投入された 1 秒当りの平均読み取り数

表 24-8 sar -b コマンドからの出力 (続き)

フィールド名	説明
lread/s	バッファークャッシュからの 1 秒当りの平均論理読み取り数
%rcache	バッファークャッシュ内で見つかった論理読み込み数の小数部 (lread/s に対する bread/s の比を 100% から差し引いた値)
bwrit/s	バッファークャッシュからディスクに書き込まれた 1 秒当りの平均物理ブロック数 (512 ブロック)
lwrit/s	バッファークャッシュへの 1 秒当りの平均論理書き込み数
%wcache	バッファークャッシュ内で見つかった論理書き込み数の小数部 (lwrit/s に対する bwrit/s の比を 100% から差し引いた値)
pread/s	キャラクタ型デバイスインタフェースを使用する 1 秒当りの平均物理読み取り数
pwrit/s	キャラクタ型デバイスインタフェースを使用する 1 秒当りの平均物理書き込み要求数

最も重要なエントリは、キャッシュヒット率 %rcache と %wcache です。この 2 つのエントリは、システムバッファリングの効率を測定します。%rcache が 90% 未満の場合や、%wcache が 65% 未満の場合は、バッファ領域を大きくすればパフォーマンスを改善できる可能性があります。

例 — バッファー動作をチェックする

次の sar -b 出力の例は、すべてのデータは許容範囲に収まっているので、%rcache バッファと %wcache バッファが処理速度低下の原因ではないことを示します。すべてのデータは許容範囲に収まっています。

```
$ sar -b
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 bread/s lread/s %rcache bwrit/s lwrit/s %wcache pread/s pwrit/s
01:00:00      0      0    100      0      0     55      0      0
02:00:02      0      0    100      0      0     55      0      0
03:00:00      0      0    100      0      0     72      0      0
04:00:00      0      0    100      0      0     56      0      0
05:00:01      0      0    100      0      0     55      0      0
06:00:00      0      0    100      0      0     55      0      0

Average      0      0     94      0      0     64      0      0
```

▼ システムコールの統計情報をチェックする方法 (sar -c)

sar -c コマンドを使用すると、システムコールの統計情報が表示されます。

```
$ sar -c
00:00:00 scall/s sread/s swrit/s fork/s exec/s rchar/s wchar/s
01:00:00      38      2      2   0.00   0.00   149   120
```

次の表は、`-c` オプションを指定したときに報告されるシステムコールのカテゴリを示します。一般に、読み取りと書き込みはシステムコール合計の約半分を占めますが、割合はシステムで実行中の動作によって大幅に変動します。

表 24-9 `sar -c` コマンドからの出力

フィールド名	説明
scall/s	1 秒当たりのすべてのタイプのシステムコール数 (通常は、4 ないし 6 ユーザーのシステム上で 1 秒当たり約 30)
sread/s	1 秒当りの read システムコール数
swrit/s	1 秒当りの write システムコール数
fork/s	1 秒当たりの fork システムコール数 (4 ないし 6 ユーザーのシステム上で毎秒約 0.5)。この数値は、シェルスクリプトの実行中は大きくなる
exec/s	1 秒当りの exec システムコール数。exec/s を fork/s で割った値が 3 より大きい場合は、効率の悪い PATH 変数を調べる
rchar/s	read システムコールによって転送される 1 秒当りの文字 (バイト) 数
wchar/s	write システムコールによって転送される 1 秒当りの文字 (バイト) 数

例 — システムコールの統計情報をチェックする

次の例に、`sar -c` コマンドからの出力を示します。

```
$ sar -c
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 scall/s sread/s swrit/s fork/s exec/s rchar/s wchar/s
01:00:00      38      2      2   0.00   0.00   149   120
02:00:02      38      2      2   0.00   0.00   149   120
03:00:00      42      2      2   0.05   0.05   218   147
04:00:00      39      2      2   0.01   0.00   155   123
05:00:01      38      2      2   0.00   0.00   150   120
06:00:00      38      2      2   0.01   0.00   149   120

Average      50      4      3   0.02   0.02   532   238
```

▼ ディスク動作をチェックする方法 (`sar -d`)

`sar -d` コマンドを使用すると、ディスク動作の統計情報が表示されます。

```

$ sar -d
00:00:00  device          %busy  avque  r+w/s  blks/s  avwait  avserv

01:00:00  fd0              0      0.0    0       0       0.0     0.0

```

次の表は、-d オプションを指定したときに報告されるディスクデバイスの動作を示します。

表 24-10 sar -d コマンドからの出力

フィールド名	説明
device	監視中のディスクデバイス名
%busy	デバイスが転送要求のサービスに費やす時間の割合
avque	平均待ち時間と平均サービス時間の合計
r+w/s	デバイスへの 1 秒当りの読み取り転送数と書き込み転送数
blks/s	デバイスに転送される 1 秒当たりの 512 バイトブロック数
avwait	待ち行列内の転送要求の平均アイドル時間 (ミリ秒単位)。これは、待ち行列が占有されている場合のみ計測される
avserv	デバイスが 1 転送要求の処理に要する平均時間 (ミリ秒単位)。ディスクの場合は、この値にシークタイム、回転待ち時間、データ転送時間が含まれる

待ち行列内に何かがあるときは、待ち行列の長さや待ち時間が計測されるので注意してください。%busy の値が小さい場合に、待ち行列とサービス時間が大きければ、変更されたブロックをディスクに随時書き込むために、システムが定期的に処理していることを示す場合があります。

例 — ディスク動作をチェックする

次の例は、sar -d コマンドからの一部省略した出力を示します。

```

$ sar -d
SunOS venus 5.9 Generic sun4u      06/24/2001

00:00:00  device          %busy  avque  r+w/s  blks/s  avwait  avserv

01:00:00  fd0              0      0.0    0       0       0.0     0.0
          nfs1              0      0.0    0       0       0.0     0.0
          sd0              0      0.0    0       0       0.0    39.6
          sd0,a           0      0.0    0       0       0.0    39.6
          sd0,b           0      0.0    0       0       0.0     0.0
          sd0,c           0      0.0    0       0       0.0     0.0
          sd0,f           0      0.0    0       0       0.0     0.0
          sd0,g           0      0.0    0       0       0.0     0.0
          sd0,h           0      0.0    0       0       0.0     0.0

```

```
sd6          0      0.0      0      0      0.0      0.0
```

▼ ページアウトとメモリーをチェックする方法 (sar -g)

sar -g コマンドを使用すると、ページアウト動作の平均値とメモリー解放動作の平均値とが表示されます。

```
$ sar -g
00:00:00  pgout/s  ppgout/s  pgfree/s  pgscan/s  %ufs_ipf
01:00:00      0.00      0.00      0.00      0.00      0.00
```

sar -g コマンドで表示される出力は、より多くのメモリーが必要かどうかを判断するのに役立ちます。ps -elf コマンドを使用すると、ページデーモンに使用される CPU サイクル数が表示されます。サイクル数が大きく、pgfree/s と pgscan/s の値が大きければ、メモリー不足を示します。

また、sar -g を使用すると、i ノードの再利用間隔が短すぎるために、再利用可能なページが失われているかどうか也表示されます。

次の表に -g オプションからの出力について説明します。

表 24-11 sar -g コマンドからの出力

フィールド名	説明
pgout/s	1 秒間にページアウトされた要求数
ppgout/s	1 秒間に実際にページアウトされたページ数(1 つのページアウト要求で複数のページがページアウトされることがある)
pgfree/s	空きリストに配置された 1 秒当りのページ数
pgscan/s	ページデーモンによって走査された 1 秒当りのページ数。この値が大きい場合は、ページデーモンが空きメモリーのチェックに大量の時間を費やしている。これは、メモリーを増やす必要があることを示す
%ufs_ipf	ufs がそれに関連付けられた再使用可能ページを持つ iget によって空きリストから取り出された割合。これらのページはフラッシュされ、プロセスが回収できなくなる。したがって、これはページフラッシュを伴う igets の割合である。値が大きければ、i ノードの空きリストがページ境界であり、ufs の i ノード数を増やす必要があることを示す

例 — ページアウトとメモリーをチェックする

次の例に、sar -g コマンドからの出力を示します。

```
$ sar -g
SunOS venus 5.9 Generic sun4u      06/24/2001
```

00:00:00	pgout/s	ppgout/s	pgfree/s	pgscan/s	%ufs_ipf
01:00:00	0.00	0.00	0.00	0.00	0.00
02:00:02	0.00	0.00	0.00	0.00	0.00
03:00:00	0.00	0.01	0.01	0.00	0.00
04:00:00	0.00	0.00	0.00	0.00	0.00
05:00:01	0.00	0.00	0.00	0.00	0.00
06:00:00	0.00	0.00	0.00	0.00	0.00
Average	0.01	0.12	0.21	0.66	0.00

▼ カーネルメモリーの割り当てをチェックする方法 (sar -k)

sar -k コマンドを使用すると、KMA (Kernel Memory Allocator) に関して次の動作が表示されます。

KMA を使用すると、カーネルサブシステムは必要に応じてメモリーを割り当て、解放できます。最大量のメモリーを静的に割り当ててるのではなく、ピークを下回る負荷を要求するのが予想されるため、KMA はメモリー要求を「小」(256 バイト未満)、「大」(512 バイト ~ 4K バイト)、「サイズ超過」(4K バイト超) という 3 つのカテゴリに分けます。また、2 つのメモリープールを管理して、「小」要求と「大」要求を満たします。「サイズ超過」要求は、システムページアロケータからメモリーを割り当てることによって満たされます。

KMA 資源を使用するドライバや STREAMS の作成に使用中のシステムを調査する場合は、sar -k コマンドを使用すると便利です。それ以外の場合は、このコマンドで提供される情報は不要です。KMA 資源を使用するが、終了前には特に資源を返さないドライバやモジュールがあると、メモリーのリークが生じることがあります。メモリーリークが発生すると、KMA によって割り当てられるメモリーは時間が経つにつれて増大します。したがって、sar -k コマンドの alloc フィールドの値が時間が経つにつれ増える場合は、メモリーリークの可能性があります。メモリーリークのもう 1 つの兆候は、要求が失敗することです。この問題が発生した場合は、メモリーリークのために KMA がメモリーを予約したり割り当てたりできなくなっている可能性があります。

メモリーリークが発生した場合は、KMA からメモリーを要求したが返していないドライバや STREAMS がないかどうかをチェックする必要があります。

```
$ sar -k
00:00:00 sml_mem  alloc  fail  lg_mem  alloc  fail  ovsz_alloc  fail
01:00:00 2523136 1866512    0 18939904 14762364    0    360448    0
02:00:02 2523136 1861724    0 18939904 14778748    0    360448    0
```

次の表に -k オプションからの出力について説明します。

表 24-12 sar -k コマンドからの出力

フィールド名	説明
sml_mem	KMA が小メモリー要求プール内で使用できるメモリーのバイト数 (小要求は 256 バイト未満)
alloc	KMA が小メモリー要求プールから小メモリー要求に割り当てたメモリーのバイト数
fail	少量のメモリーで失敗した要求数
lg_mem	KMA が大メモリー要求プール内で使用できるメモリーのバイト数 (大要求は 512 バイトから 4K バイトまで)
alloc	KMA が大メモリー要求プールから大メモリー要求に割り当てたメモリーのバイト数
fail	大メモリーで失敗した要求数
ovsz_alloc	サイズ超過要求 (4K バイトを超える要求) に割り当てられたメモリーの容量。これらの要求はページロケータによって満たされる。このため、プールはない
fail	サイズ超過メモリーで失敗した要求数

例 — カーネルメモリーの割り当てをチェックする (sar)

次の例は、sar -k 出力を示します。

```
$ sar -k
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 sml_mem  alloc  fail  lg_mem  alloc  fail  ovsz_alloc  fail
01:00:00 2523136 1866512    0 18939904 14762364    0    360448    0
02:00:02 2523136 1861724    0 18939904 14778748    0    360448    0
03:00:00 2523136 1865664    0 18939904 14745884    0    360448    0
04:00:00 2523136 1867692    0 18939904 14746616    0    360448    0
05:00:01 2523136 1867208    0 18939904 14763700    0    360448    0
06:00:00 2523136 1867772    0 18939904 14779444    0    360448    0

Average 2724096 1791806    0 20089344 15434591    0    360448    0
```

▼ プロセス間通信をチェックする方法 (sar -m)

sar -m コマンドを使用すると、プロセス間通信の動作が表示されます。

```
$ sar -m
00:00:00  msg/s  sema/s
01:00:00   0.00   0.00
```

通常、これらの数字は、メッセージやセマフォを使用するアプリケーションを実行していない限りゼロ (0.00) です。

次の表に -m オプションからの出力について説明します。

表 24-13 sar -m コマンドからの出力

フィールド名	説明
msg/s	1 秒当りのメッセージ処理 (送受信) 数
sema/s	1 秒当りのセマフォ処理数

例 — プロセス間通信をチェックする

次の例は、sar -m コマンドからの一部省略した出力を示します。

```
$ sar -m
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00  msg/s  sema/s
01:00:00    0.00    0.00
02:00:02    0.00    0.00
03:00:00    0.00    0.00
04:00:00    0.00    0.00
05:00:01    0.00    0.00
06:00:00    0.00    0.00

Average    0.00    0.00
```

▼ ページイン動作をチェックする方法 (sar -p)

sar -p コマンドを使用すると、保護フォルトや変換フォルトを含むページイン動作が表示されます。

```
$ sar -p
00:00:00  atch/s  pgin/s  ppgin/s  pflt/s  vflt/s  slock/s
01:00:00    0.07    0.00    0.00    0.21    0.39    0.00
```

次の表に -p オプションから報告される統計情報を示します。

表 24-14 sar -p コマンドからの出力

フィールド名	説明
atch/s	現在メモリーに入っているページを回収して満たされる 1 秒当りのページフォルト数 (1 秒当りの付加数)。この例には、空きリストから無効なページを回収し、別のプロセスに現在使用中のテキストページを共有する処理が含まれる (たとえば、複数のプロセスが同じプログラムテキストにアクセスしている場合など)
pgin/s	ファイルシステムがページイン要求を受信する 1 秒当りの回数

表 24-14 sar -p コマンドからの出力 (続き)

フィールド名	説明
ppgin/s	ページインされる 1 秒当りのページ数。ソフトロック要求 (slock/s を参照) などの 1 つのページイン要求や、大型ブロックサイズでは、複数のページがページインされることがある
pflt/s	保護エラーによるページフォルト数。保護フォルトの例には、ページへの不正なアクセスや、「書き込み時コピー」などがある。通常、この数値は主に「書き込み時コピー」からなっている
vflt/s	1 秒当りのアドレス変換ページフォルト数。これは、有効性フォルトと呼ばれ、所定の仮想アドレスに有効なプロセステーブルエントリが存在しないときに発生する
slock/s	物理入出力を要求するソフトウェアロック要求によって発生する 1 秒当りのフォルト数。ソフトロック要求の発生例には、ディスクからメモリーへのデータ転送などがある。システムはデータを受信しないページをロックするので、別のプロセスはそれを回収して使用できない

例 — ページイン動作をチェックする

次の例に、sar -g コマンドからの出力を示します。

```
$ sar -p
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00  atch/s  pgin/s  ppgin/s  pflt/s  vflt/s  slock/s
01:00:00   0.07   0.00   0.00    0.21   0.39   0.00
02:00:02   0.07   0.00   0.00    0.21   0.39   0.00
03:00:00   0.32   0.00   0.00    1.10   2.48   0.00
04:00:00   0.09   0.00   0.00    0.32   0.57   0.00
05:00:01   0.07   0.00   0.00    0.21   0.39   0.00
06:00:00   0.07   0.00   0.00    0.21   0.39   0.00

Average    0.26    0.20    0.30    0.92    1.78    0.00
```

▼ 待ち行列動作をチェックする方法 (sar -q)

sar -q コマンドを使用すると、待ち行列に要求が入っている平均待ち行列の長さ、その間の時間の割合が表示されます。

```
$ sar -q
00:00:00  runq-sz  %runocc  swpq-sz  %swpocc
```

注 - システムに空きメモリーが十分あっても、スワップアウトされた軽量プロセス数が 0 より大きい場合があります。この状態は、休眠中の軽量プロセスがスワップアウトされ処理されない場合に発生します。たとえば、プロセスや軽量プロセスが休眠中であり、キーボードやマウスの入力を待機中の場合などです。

次の表に `-q` オプションからの出力について説明します。

表 24-15 `sar -q` コマンドの出力

フィールド名	説明
<code>runq-sz</code>	CPU を実行するためにメモリー内で待機中のカーネルスレッド数。通常、この値は 2 未満になる。値が常に 2 より大きい場合は、システムが CPU の限界に到達している可能性がある
<code>%runocc</code>	ディスクパッチ待ち行列が使用されている時間の割合
<code>swpq-sz</code>	スワップアウトされた平均軽量プロセス数
<code>%swpocc</code>	軽量プロセスがスワップアウトされた時間の割合

例 — 待ち行列動作をチェックする

次の例は、`sar -q` コマンドからの一部省略した出力を示します。`%runocc` の値が大きく (90 パーセント超)、`runq-sz` が 2 より大きい場合は、CPU の負荷が大きく、応答速度が低下しています。この場合は、CPU の容量を増やしてシステムの応答速度を適正化する必要があります。

```
$ sar -q
SunOS venus 5.9 Generic sun4u    06/24/2001

08:45:18 runq-sz %runocc swpq-sz %swpocc
08:45:18          unix restarts
09:00:00      1.0          0      0.0          0
09:20:00      0.0          0      0.0          0
09:40:00      0.0          0      0.0          0

Average      1.0          0      0.0          0
```

▼ 未使用のメモリーをチェックする方法 (`sar -r`)

`sar -r` コマンドを使用すると、現在使用されていないメモリーページ数とスワップファイルのディスクブロック数が表示されます。

```
$ sar -r
00:00:00 freemem freeswap
01:00:00    2135    401922
```

次の表に `-r` オプションからの出力について説明します。

表 24-16 `sar -r` コマンドからの出力

フィールド名	説明
<code>freemem</code>	コマンドによるサンプル収集間隔の間にユーザープロセスに利用できる平均メモリーページ数。ページサイズはマシンに応じて異なる
<code>freeswap</code>	ページスワップに使用可能な 512 バイトのディスクブロック数

例 — 未使用のメモリーをチェックする

次の例は、`sar -r` コマンドからの出力を示します。

```
$ sar -r
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 freemem freeswap
01:00:00    2135    401922
02:00:02    2137    401949
03:00:00    2137    402006
04:00:00    2139    401923
05:00:01    2138    402033
06:00:00    2137    401919

Average    2500    399914
```

▼ CPU の使用状況をチェックする方法 (`sar -u`)

`sar -u` コマンドを使用すると、CPU 使用状況の統計情報が表示されます。

```
$ sar -u
00:00:00    %usr    %sys    %wio    %idle
01:00:00         0         0         0    100
```

オプションを指定しない `sar` コマンドは、`sar -u` と同じです。特定の瞬間では、プロセッサはビジー状態またはアイドル状態になっています。ビジー状態のときは、プロセッサはユーザーモードまたはシステムモードになっています。アイドル状態のときは、プロセッサは入出力の完了を待っているか、何も処理することがないので「待機」している状態です。

次の表に `-u` オプションからの出力について説明します。

表 24-17 sar -u コマンドからの出力

フィールド名	説明
%usr	プロセッサがユーザーモードになっている時間の割合が表示される
%sys	プロセッサがシステムモードになっている時間の割合が表示される
%wio	プロセッサがアイドル状態で入出力の完了を待っている時間の割合が表示される
%idle	プロセッサがアイドル状態で入出力を待っていない時間の割合が表示される

一般に、%wio の値が大きい場合は、ディスクの処理速度が低下していることを意味します。

例 — CPU の使用状況をチェックする

次の例は、sar -u コマンドからの出力を示します。

```
$ sar -u
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00   %usr   %sys   %wio   %idle
01:00:00     0     0     0    100
02:00:02     0     0     0    100
03:00:00     0     0     0    100
04:00:00     0     0     0    100
05:00:01     0     0     0    100
06:00:00     0     0     0    100
07:00:00     0     0     0    100
08:00:01     0     0     0    100
08:20:00     0     0     0    100
08:40:00     0     0     0    100
09:00:00     0     0     0    100
09:20:00     0     0     0    100
09:40:00     0     0     0    100
10:00:00     0     0     0    100
10:20:00     0     0     0    100
10:40:01     0     0     0    100
11:00:00     5     2    10     82

Average     0     0     0    100
```

▼ システムテーブルの状態をチェックする方法 (sar -v)

sar -v コマンドを使用すると、プロセステーブル、i ノードテーブル、ファイルテーブル、および共有メモリーレコードテーブルの状態が表示されます。

```
$ sar -v
00:00:00 proc-sz   ov inod-sz   ov file-sz   ov lock-sz
01:00:00  43/922     0 2984/4236  0 322/322   0  0/0
```

次の表に -v オプションからの出力について説明します。

表 24-18 sar -v コマンドからの出力

フィールド名	説明
proc-sz	現在カーネル内で使用されているか、割り当てられているプロセスエントリ (proc 構造体) の数
inod-sz	メモリー内の合計 i ノード数とカーネル内で割り当て済みの最大 i ノード数の比。これは厳密な上限ではなく、超えることもできる
file-sz	開いているシステムファイルテーブルのサイズ。ファイルテーブルには領域が動的に割り当てられるので、sz は 0 として表示される
ov	現在カーネル内で使用されているか割り当てられている共有メモリーレコードテーブルのエントリ数。共有メモリーレコードテーブルには領域が動的に割り当てられるので、sz は 0 として表示される
lock-sz	現在カーネル内で使用されているか割り当てられている共有メモリーレコードテーブルのエントリ数。共有メモリーレコードテーブルには領域が動的に割り当てられるので、sz は 0 として表示される

例 — システムテーブルの状態をチェックする

次の例は、sar -v コマンドからの一部省略した出力を示します。この例は、すべてのテーブルに十分なサイズがあり、オーバーフローは発生しないことを示します。これらのテーブルには、いずれも物理メモリーの容量に基づいて領域が動的に割り当てられます。

```
$ sar -v
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 proc-sz   ov inod-sz   ov file-sz   ov lock-sz
01:00:00  43/922     0 2984/4236  0 322/322   0  0/0
02:00:02  43/922     0 2984/4236  0 322/322   0  0/0
03:00:00  43/922     0 2986/4236  0 323/323   0  0/0
04:00:00  43/922     0 2987/4236  0 322/322   0  0/0
05:00:01  43/922     0 2987/4236  0 322/322   0  0/0
```

```
06:00:00 43/922 0 2987/4236 0 322/322 0 0/0
```

▼ スワップ動作をチェックする方法 (sar -w)

sar -w コマンドを使用すると、スワッピングと切り替え動作が表示されます。

```
$ sar -w
00:00:00 swpin/s bswin/s swpot/s bswot/s pswch/s
01:00:00 0.00 0.0 0.00 0.0 22
```

次の表に、対象となる値と説明を示します。

表 24-19 sar -w コマンドからの出力

フィールド名	説明
swpin/s	メモリーに転送される 1 秒当りの軽量プロセス数
bswin/s	メモリーからスワップアウトされる 1 秒当りの平均プロセス数。 この数値が 1 より大きい場合は、メモリーを増やす必要がある
swpot/s	メモリーからスワップアウトされる 1 秒当りの平均プロセス数。 この数値が 1 より大きい場合は、メモリーを増やす必要がある
bswot/s	スワップアウト用に転送される 1 秒当りのブロック数
pswch/s	1 秒当りのカーネルスレッド切り替え数

注 - すべてのプロセスのスワップインには、プロセスの初期化が含まれます。

例 — スワップ動作をチェックする

次の例は、sar -w コマンドからの出力を示します。

```
$ sar -w
SunOS venus 5.9 Generic sun4u 06/24/2001

00:00:00 swpin/s bswin/s swpot/s bswot/s pswch/s
01:00:00 0.00 0.0 0.00 0.0 22
02:00:02 0.00 0.0 0.00 0.0 22
03:00:00 0.00 0.0 0.00 0.0 22
04:00:00 0.00 0.0 0.00 0.0 22
05:00:01 0.00 0.0 0.00 0.0 22
06:00:00 0.00 0.0 0.00 0.0 22
07:00:00 0.00 0.0 0.00 0.0 22
08:00:01 0.00 0.0 0.00 0.0 22
08:20:00 0.00 0.0 0.00 0.0 22
08:40:00 0.00 0.0 0.00 0.0 22
```



```

09:00:00    0.00    0.0    0.00    0.0    22
09:20:00    0.00    0.0    0.00    0.0    22
09:40:00    0.00    0.0    0.00    0.0    22
10:00:00    0.00    0.0    0.00    0.0    22
10:20:00    0.00    0.0    0.00    0.0    22
10:40:01    0.00    0.0    0.00    0.0    23
11:00:00    0.00    0.0    0.00    0.0    144

Average    0.00    0.0    0.00    0.0    24

```

▼ 端末動作をチェックする方法 (sar -y)

sar -y コマンドを使用すると、端末デバイスの動作を監視できます。

```

$ sar -y
00:00:00 rawch/s canch/s outch/s rcvin/s xmtin/s mdmin/s
01:00:00      0      0      0      0      0      0

```

大量の端末入出力がある場合は、このレポートを使用して不良な行がないかどうかを判別できます。次の表に、記録される動作を示します。

表 24-20 sar -y コマンドからの出力

フィールド名	説明
rawch/s	1 秒当りの入力文字数 (raw 待ち行列)
canch/s	標準待ち行列で処理される 1 秒当りの文字数
outch/s	1 秒当りの出力文字数 (出力待ち行列)
rcvin/s	1 秒当りの受信側ハードウェア割り込み数
xmtin/s	1 秒当りの送信側ハードウェア割り込み数
mdmin/s	1 秒当りのモデム割り込み数

1 秒当りのモデム割り込み数 (mdmin/s) は、0 に近い値になります。また、1 秒当りの送受信側ハードウェア割り込み数 (xmtin/s と rcvin/s) は、それぞれ着信または発信文字数以下になります。そうでない場合は、不良回線がないかどうかをチェックしてください。

例 — 端末動作をチェックする

次の例は、sar -y コマンドからの一部省略した出力を示します。

```

$ sar -y
SunOS venus 5.9 Generic sun4u    06/24/2001

00:00:00 rawch/s canch/s outch/s rcvin/s xmtin/s mdmin/s

```

01:00:00	0	0	0	0	0	0
02:00:02	0	0	0	0	0	0
03:00:00	0	0	0	0	0	0
04:00:00	0	0	0	0	0	0
05:00:01	0	0	0	0	0	0
06:00:00	0	0	0	0	0	0
07:00:00	0	0	0	0	0	0
08:00:01	0	0	0	0	0	0
08:20:00	0	0	0	0	0	0
08:40:00	0	0	0	0	0	0
09:00:00	0	0	0	0	0	0
09:20:00	0	0	0	0	0	0
09:40:00	0	0	0	0	0	0
10:00:00	0	0	0	0	0	0
10:20:00	0	0	0	0	0	0
10:40:01	0	0	20	0	0	0
Average	0	0	3	0	0	0

▼ システム全体のパフォーマンスをチェックする方法 (sar -A)

sar -A コマンドを使用すると、すべてのオプションを指定した場合と同じように、システム全体を示す統計情報が表示されます。

このコマンドを使用すると、全体像を把握できます。複数のタイムセグメントからのデータが表示される場合は、レポートに平均値が含まれます。

システム動作データの自動収集 (sar)

システム動作データを自動的に収集するには、sadc、sa1、sa2 という3つのコマンドを使用します。

sadc データ収集ユーティリティは、システム動作に関するデータを定期的に収集し、24時間ごとに1つのファイルに2進形式で保存します。sadc コマンドを定期的に (通常は1時間ごとに) 実行したり、システムがマルチユーザーモードでブートするときにも実行するように設定できます。データファイルは、/var/adm/sa ディレクトリに格納されます。各ファイルには sadd という名前が与えられます。この場合、dd は現在の日付です。このコマンドの書式は次のとおりです。

```
/usr/lib/sa/sadc [t n] [ofile]
```

このコマンドは、t 秒 (5秒より長くする必要があります) 間隔でサンプルデータを n 回収集します。次に、2進形式 ofile ファイルまたは標準出力に書き込みます。

ブート時に `sadc` コマンドを実行する

カウンタが0にリセットされるときから統計情報を記録するために、`sadc` コマンドをシステムのブート時に実行する必要があります。`sadc` をブート時に確実に実行するために、日ごとのデータファイルにレコードを書き込むコマンド行が `/etc/init.d/perf` ファイルに含まれます。

コマンドエントリの書式は次のとおりです。

```
/usr/bin/su sys -c "/usr/lib/sa/sadc /var/adm/sa/sàdate +%d`"
```

sa1 スクリプトを使用して `sadc` コマンドを定期的に実行する

定期的にレコードを生成するには、`sadc` コマンドを定期的に実行する必要があります。そのためには、`/var/spool/cron/crontab/sys` ファイルの次の行をコメント解除するのが最も簡単な方法です。

```
# 0 * * * 0-6 /usr/lib/sa/sa1
# 20,40 8-17 * * 1-5 /usr/lib/sa/sa1
# 5 18 * * 1-5 /usr/lib/sa/sa2 -s 8:00 -e 18:01 -i 1200 -A
```

`sys` の `crontab` エントリによって、次のように動作します。

- 最初の2つの `crontab` エントリによって、月曜から金曜までは午前8時から午後5時まで20分ごとに、それ以外の曜日は1時間ごとに、レコードが `/var/adm/sa/sa dd` に書き込まれます。
- 3番目のエントリは、月曜から金曜までは1時間ごとに、レコードを `/var/adm/sa/sar dd` ファイルに書き込み、すべての `sar` オプションが含まれます。

これらのデフォルトは、必要に応じて変更できます。

sa2 スクリプトを使用してレポートを生成する

もう1つのシェルスクリプト `sa2` は、2進データファイルではなくレポートを生成します。`sa2` コマンドは `sar` コマンドを呼び出して、レポートファイルにASCII出力を書き込みます。

データの自動収集を設定する (`sar`)

`sar` コマンドを使用すると、システム動作データそのものを収集するか、`sadc` コマンドで作成された日ごとの動作ファイルに収集された情報をレポートできます。

`sar` コマンドの書式は次のとおりです。

```
sar [-aAbcdgkmpqruvwy] [-o file] t [n]
```

```
sar [-aAbcdgkmpqruvwy] [-s time] [-e time] [-i sec] [-f file]
```

次の `sar` コマンドは、オペレーティングシステム内の累積動作カウンタから t 秒間隔で n 回データを収集します。 t は、5 秒以上の値にします。それ以外の値にすると、コマンドそのものがサンプルに影響を与えることがあります。また、サンプルの収集間隔を指定する必要があります。指定しないと、このコマンドは第 2 の書式に従って動作します。 n のデフォルト値は 1 です。次の例では、10 秒間隔で 2 つのサンプルが収集されます。`-o` オプションを指定すると、サンプルは 2 進形式でファイルに保存されます。

```
$ sar -u 10 2
```

その他に、`sar` では次の点に注意する必要があります。

- サンプル間隔またはサンプル数を指定しなければ、`sar` コマンドはデータを以前に記録されたファイルから抽出します。その場合は、`-f` オプションで指定したファイル、またはデフォルトでは最新日付分の標準の日ごとの動作ファイル `/var/adm/sa/sadd` から抽出されます。
- `-s` オプションと `-e` オプションでは、レポートの開始時刻と終了時刻を定義します。開始時刻と終了時刻の書式は `hh[:mm[:ss]]` です (この場合、 h 、 m 、 s は、それぞれ時間、分、秒を表します)。
- `-i` オプションでは、レコードの選択間隔を秒単位で指定します。`-i` オプションを指定しなければ、日ごとの動作ファイル内で見つかったすべての間隔がレポートされます。

次の表に `sar` コマンドのオプションとその動作を示します。

表 24-21 `sar` コマンドのオプション

オプション	動作
<code>-a</code>	ファイルアクセス操作をチェックする
<code>-b</code>	バッファー動作をチェックする
<code>-c</code>	システムコールをチェックする
<code>-d</code>	各ブロックデバイスの動作をチェックする
<code>-g</code>	ページアウトとメモリーの解放をチェックする
<code>-k</code>	カーネルメモリーの割り当てをチェックする
<code>-m</code>	プロセス間通信をチェックする
<code>-p</code>	スワップとディスパッチ動作をチェックする
<code>-q</code>	待ち行列動作をチェックする
<code>-r</code>	未使用メモリーをチェックする

表 24-21 sar コマンドのオプション (続き)

オプション	動作
-u	CPU の使用率をチェックする
-nv	システムテーブルの状態をチェックする
-w	ボリュームのスワッピングと切り替えをチェックする
-y	端末動作をチェックする
-A	システム全体のパフォーマンスをリポートする (すべてのオプションを入力した場合と同じです)

オプションを使用しなければ、-u オプションを指定してコマンドを呼び出すのと同じです。

自動データ収集を設定する方法

1. スーパーユーザーになります。
2. /etc/init.d/perf ファイルを編集して次の行をコメント解除します。

```
# if [ -z "$_INIT_RUN_LEVEL" ]; then
#     set -- ` /usr/bin/who -r `
#     _INIT_RUN_LEVEL="$7"
#     _INIT_RUN_NPREV="$8"
#     _INIT_PREV_LEVEL="$9"
# fi
#
# if [ $_INIT_RUN_LEVEL -ge 2 -a $_INIT_RUN_LEVEL -le 4 -a \
#     $_INIT_RUN_NPREV -eq 0 -a \( $_INIT_PREV_LEVEL = 1 -o \
#     $_INIT_PREV_LEVEL = S \) ]; then
#
#     /usr/bin/su sys -c "/usr/lib/sa/sadc /var/adm/sa/sàdate +%d`"
# fi
```

このバージョンの sadc コマンドは、カウンタが 0 にリセットされる時間 (ブート時) を示す特殊なレコードを書き込みます。

3. crontab ファイル /var/spool/cron/crontabs/sys の次の行をコメント解除します。

```
# 0 * * * 0-6 /usr/lib/sa/sa1
# 20,40 8-17 * * 1-5 /usr/lib/sa/sa1
# 5 18 * * 1-5 /usr/lib/sa/sa2 -s 8:00 -e 18:01 -i 1200 -A
```


第 25 章

Solaris ソフトウェアで発生する問題の解決 (トピック)

以下の各章で、Solaris ソフトウェアで発生する問題を解決する手順を説明します。

第 26 章	一般的なソフトウェアの問題を解決する方法の概要と、システムクラッシュを解決する手順を説明します。
第 27 章	コアファイルを管理する手順を説明します。
第 28 章	クラッシュダンプを保存する手順とシステムエラー記録をカスタマイズする手順を説明します。
第 29 章	一般的なソフトウェアの問題 (システムがハングする、システムがブートしないなど) の状況と可能な解決策を説明します。
第 30 章	一般的なファイルアクセスについての問題 (正しくないコマンド検索パスやファイルのアクセス権など) の解決策を説明します。
第 31 章	一般的なプリンタの問題 (出力が出ない、出力が正しくないなど) の解決策を説明します。
第 32 章	UFS ファイルシステム以外の問題について、特定の fsck エラーメッセージとその解決策を説明します。
第 33 章	ソフトウェアパッケージを追加および削除するときに発生する問題について、特定のエラーメッセージと可能な解決策を説明します。

第 26 章

ソフトウェアの問題解決 (概要)

この章では、ソフトウェアの問題の解決についての概要を説明します。システムクラッシュの問題の解決とシステムメッセージの表示などが含まれます。

この章の内容は次のとおりです。

- 361 ページの「ソフトウェアの問題解決に関する新機能」
- 363 ページの「ソフトウェアの問題の解決方法の参照先」
- 363 ページの「システムクラッシュの問題の解決」
- 365 ページの「システムクラッシュを解決するためのチェックリスト」
- 366 ページの「システムメッセージの表示」
- 368 ページの「システムのメッセージ記録のカスタマイズ」

ソフトウェアの問題解決に関する新機能

この節では、Solaris 9 リリースでの新機能について説明します。

新しいシステムログローテーション

Solaris 9 リリースでは、システムログファイルローテーションが、ルートの `crontab` ファイルのエントリから `logadm` コマンドによって実行されます。
`/usr/lib/newsyslog` スクリプトは使用されません。

この新しいシステムログローテーションは、`/etc/logadm.conf` ファイルに定義されます。このファイルには、`syslogd` などのプロセスのログローテーションエントリが含まれています。たとえば、`/etc/logadm.conf` ファイルにある 1 つのエントリは、`/var/log/syslog` ファイルが空でなければ、そのローテーションが毎週実行されます。つまり、最新の `syslog` ファイルが `syslog.0` になり、その次に新しい `syslog` ファイルが `syslog.1` になります。最新からさかのぼって 8 つまでの `syslog` ログファイルが保存されます。

また、`/etc/logadm.conf` ファイルには、最後のログローテーション実行時のタイムスタンプも含まれます。

`logadm` コマンドを使用して、必要に応じてシステムログをカスタマイズしたり、`/etc/logadm.conf` ファイルにログを追加したりすることができます。

たとえば、Apache アクセスとエラーログのローテーションを実行するには、次のコマンドを使用します。

```
# logadm -w /var/apache/logs/access_log -s 100m
# logadm -w /var/apache/logs/error_log -s 10m
```

この例では、Apache の `access_log` ファイルのローテーションは、そのサイズが 100 MB に達したときに実行され、そのファイル名に `.0`、`.1` などのように接尾辞が付けられます。また、古い `access_log` ファイルのコピーが 10 個保存されます。また、`error_log` のローテーションは、そのサイズが 10 MB に達したときに実行され、`access_log` ファイルと同様に、接尾辞が付けられ、コピーが作成されます。

前述の Apache ログローテーションの例における `/etc/logadm.conf` エントリの例は、次のようになります。

```
# cat /etc/logadm.conf
.
.
.
/var/apache/logs/error_log -s 10m
/var/apache/logs/access_log -s 100m
```

詳細については、`logadm(1M)` のマニュアルページを参照してください。

スーパーユーザーでログインするか、同等の役割 (ログ管理の権限を持つ) でアクセスすることによって、`logadm` コマンドを使用できます。役割によるアクセス制御 (RBAC) を設定すれば、`logadm` コマンドへのアクセス権を与えることで、ルート以外のユーザーにログ管理の権限を与えることができます。

たとえば、次のエントリを `/etc/user_attr` ファイルに追加して、`logadm` コマンドを使用できる `andy` をユーザーに与えます

```
andy:::profiles=Log Management
```

または、Solaris 管理コンソールを使用して、ログ管理の役割を設定できます。役割の設定に関する詳細については、『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」を参照してください。

新しい root アカウトに対するシェルのフォールバック

Solaris の前のリリースでは、`root` のシェルを存在しないシェルに変更すると、`CD` またはネットワークからシステムをブートし、`/etc/passwd` ファイルにある `root` シェルを修正する必要がありました。

Solaris 9 リリースでは、root に存在しないシェルを間違えて入力しても、次のいずれかを実行すると、root のシェルが自動的に /sbin/sh にフォールバックされます。

- su コマンドを使用して、スーパーユーザー (root) のアカウントになる。
- シングルユーザーモードのプロンプトで root のパスワードを入力する。

詳細については、su (1M) のマニュアルページを参照してください。

ソフトウェアの問題の解決方法の参照先

問題解決の手順	参照箇所
システムクラッシュ情報の管理	第 28 章
コアファイルの管理	第 27 章
リポート失敗、バックアップ問題などのソフトウェアの問題解決	第 29 章
ファイルアクセスの問題解決	第 30 章
印刷の問題解決	第 31 章
UFS ファイルシステムの不整合の解決	第 32 章
ソフトウェアパッケージの問題解決	第 33 章

システムクラッシュの問題の解決

Solaris オペレーティング環境が動作しているシステムがクラッシュした場合は、クラッシュダンプファイルを含む、可能なかぎりの情報を購入先に提供してください。

システムがクラッシュした場合の対処方法

最も重要なことは、次のとおりです。

1. システムのコンソールメッセージを書き取ります。

システムがクラッシュした場合は、システムをリポートする前に、まずコンソール画面にメッセージが表示されていないか確認してください。このようなメッセージは、クラッシュした原因を解明するのに役立ちます。システムが自動的にリポート

して、コンソールメッセージが画面から消えた場合でも、システムエラーログファイル `/var/adm/messages` を表示すれば、これらのメッセージをチェックできます。システムエラーログファイルを表示する方法の詳細は、367 ページの「システムメッセージを表示する方法」を参照してください。

クラッシュが頻繁に発生して、その原因を特定できない場合は、システムのコンソールや `/var/adm/messages` ファイルから得られるすべての情報を収集して、購入先に問い合わせてください。購入先に問い合わせるときに必要な問題解決のための情報の完全なリストについては、363 ページの「システムクラッシュの問題の解決」を参照してください。

システムのクラッシュ後にリポートが失敗する場合は、第 29 章を参照してください。

2. 次のように入力してディスクとの同期をとり、リポートします。

```
ok sync
```

システムのクラッシュ後にリポートが失敗する場合は、第 29 章を参照してください。

また、システムクラッシュダンプがシステムのクラッシュ後に生成されたかどうかを確認してください。デフォルトでは、システムクラッシュダンプが保存されます。クラッシュダンプの詳細については、第 28 章を参照してください。

問題の解決に使用するデータの収集

システムの問題を特定するために、次の質問に答えてください。クラッシュしたシステムの問題を解決するためのデータを収集するには、365 ページの「システムクラッシュを解決するためのチェックリスト」を参照してください。

表 26-1 システムクラッシュに関するデータの収集

質問	説明
問題を再現できるか	この質問は、再現可能なテストケースは実際のハードウェア問題をデバッグするために重要であることが多いために重要である。購入先では、特殊な計測機構を使用してカーネルを構築して問題を再現し、バグを引き起こし、診断、および修正できる
<i>Sun</i> 以外のドライバを使用しているか	ドライバは、カーネルと同じアドレス空間で、カーネルと同じ特権で動作する。したがって、ドライバにバグがあると、システムクラッシュの原因となることがある
クラッシュの直前にシステムは何を実行していたか	システムが通常でないこと(新しい負荷テストの実行など)を行なったり、通常よりも高い負荷がシステムにかかったりした場合、クラッシュの原因となることがある

表 26-1 システムクラッシュに関するデータの収集 (続き)

質問	説明
クラッシュ直前に、異常なコンソールメッセージが表示されたか	システムは、実際にクラッシュする前に問題の兆候を示すことがある。この情報は役立つことが多い
/etc/system ファイルに調整パラメータを追加したか	調整パラメータは、システムクラッシュの原因となることがある。たとえば、共有メモリーセグメントを増やした結果、システムが限度以上の多くのメモリーを割り当てようとした
問題は最近発生するようになったか	そうであれば、問題の原因は、システムの変更 (たとえば、新しいドライバ、新しいソフトウェア、作業負荷の変化、CPU のアップグレード、メモリーのアップグレードなど) にある可能性がある

システムクラッシュを解決するための チェックリスト

クラッシュしたシステムの問題を解決するためのデータを収集するときは、次のチェックリストを使用します。

項目	ユーザーのデータ
システムクラッシュダンプがあるか	
オペレーティングシステムのリリースと適切なソフトウェアアプリケーションのリリースレベルを確認する	
システムのハードウェアを確認する	
sun4u システムの prtdiag 出力を含める他のシステムの Explorer 出力を含める	
パッチはインストールされているか。そうであれば、showrev -p 出力を含める	
問題を再現できるか	
Sun 以外のドライバをシステムで使用しているか	
クラッシュ直前のシステムの動作は	
クラッシュ直前に、異常なコンソールメッセージが表示されたか	

項目	ユーザーのデータ
/etc/system ファイルにパラメータを追加したか	
問題は最近発生するようになったか	

システムメッセージの表示

システムのメッセージはコンソールデバイスに表示されます。ほとんどのシステムメッセージは次の形式で表示されます。

[ID *msgid facility.priority*]

次に例を示します。

```
[ID 672855 kern.notice] syncing file systems...
```

カーネルから出されるメッセージには、カーネルモジュール名が次のように表示されます。次に例を示します。

```
Oct 1 14:07:24 mars ufs: [ID 845546 kern.notice] alloc: /: file system full
```

システムがクラッシュすると、システムのコンソールに次のようなメッセージが表示されることがあります。

```
panic: error message
```

パニックメッセージより頻度は少ないですが、パニックメッセージではなく次のメッセージが表示されることがあります。

```
Watchdog reset !
```

エラー記録デーモン `syslogd` は、自動的に様々なシステムの警告やエラーをメッセージファイルに記録します。デフォルトでは、これらのシステムメッセージの多くは、システムコンソールに表示されて、`/var/adm` ディレクトリに格納されます。システムメッセージ記録を設定することによって、これらのメッセージを格納する場所を指示できます。詳細については、369 ページの「システムのメッセージ記録をカスタマイズする方法」を参照してください。これらのメッセージは、失敗の予兆のあるデバイスなど、システム障害をユーザーに警告できます。

`/var/adm` ディレクトリには、いくつかのメッセージファイルが含まれています。最も新しいメッセージは、`/var/adm/messages` (および `messages.*`) にあり、最も古いメッセージは、`messages.3` にあります。一定の期間 (通常は 10 日) ごとに、新しい `messages` ファイルが作成されます。`messages.0` のファイル名は `messages.1` に、`messages.1` は `messages.2` に、`messages.2` は `messages.3` にそれぞれ変更されます。その時点の `/var/adm/messages.3` は削除されます。

/var/adm ディレクトリは、メッセージやクラッシュダンプなどのデータを含んでいる大きなファイルを格納するため、多くのディスク容量を消費します。/var/adm ディレクトリが大きくなるようにするために、そして将来のクラッシュダンプが保存できるようにするために、不要なファイルを定期的に削除しなければなりません。crontab ファイルを使用すれば、この作業は自動化できます。この作業の自動化の詳細については、239 ページの「クラッシュダンプファイルを削除する方法」と第 18 章を参照してください。

▼ システムメッセージを表示する方法

システムクラッシュまたはリポートによって生成された最近のメッセージを表示するには、dmesg コマンドを使用します。

```
$ dmesg
```

あるいは、more コマンドを使用して、メッセージを 1 画面ごとに表示します。

```
$ more /var/adm/messages
```

詳細については、dmesg(1M) のマニュアルページを参照してください。

例 — システムメッセージを表示する

次の例は、dmesg コマンドからの出力を示しています。

```
$ dmesg
Jan  3 08:44:41 starbug genunix: [ID 540533 kern.notice] SunOS Release 5.9 ...
Jan  3 08:44:41 starbug genunix: [ID 913631 kern.notice] Copyright 1983-2002 ...
Jan  3 08:44:41 starbug genunix: [ID 678236 kern.info] Ethernet address ...
Jan  3 08:44:41 starbug unix: [ID 389951 kern.info] mem = 131072K (0x8000000)
Jan  3 08:44:41 starbug unix: [ID 930857 kern.info] avail mem = 121888768
Jan  3 08:44:41 starbug rootnex: [ID 466748 kern.info] root nexus = Sun Ultra 5/
10 UPA/PCI (UltraSPARC-IIi 333MHz)
Jan  3 08:44:41 starbug rootnex: [ID 349649 kern.info] pcipsy0 at root: UPA 0x1f0x0
Jan  3 08:44:41 starbug genunix: [ID 936769 kern.info] pcipsy0 is /pci@1f,0
Jan  3 08:44:41 starbug pcipsy: [ID 370704 kern.info] PCI-device: pci@1,1, simba0
Jan  3 08:44:41 starbug genunix: [ID 936769 kern.info] simba0 is /pci@1f,0/pci@1,1
Jan  3 08:44:41 starbug pcipsy: [ID 370704 kern.info] PCI-device: pci@1, simba1
Jan  3 08:44:41 starbug genunix: [ID 936769 kern.info] simba1 is /pci@1f,0/pci@1
Jan  3 08:44:57 starbug simba: [ID 370704 kern.info] PCI-device: ide@3, uata0
Jan  3 08:44:57 starbug genunix: [ID 936769 kern.info] uata0 is /pci@1f,0/pci@1,
1/ide@3
Jan  3 08:44:57 starbug uata: [ID 114370 kern.info] dad0 at pci1095,6460
.
.
.
```

システムのメッセージ記録のカスタマイズ

/etc/syslog.conf ファイルを変更すると、様々なシステムプロセスが生成するエラーメッセージを記録できます。デフォルトでは、/etc/syslog.conf は、多くのシステムプロセスのメッセージが /var/adm/messages ファイルに格納されるように指示します。クラッシュとブートのメッセージも、同様にこのファイルに格納されます。/var/adm メッセージを表示する方法については、367 ページの「システムメッセージを表示する方法」を参照してください。

/etc/syslog.conf ファイルは、タブで区切られた 2 つの列から構成されています。

facility.level ... action

facility.level

機能またはメッセージや状態のシステムでの出所。コマンドで区切られた機能のリスト。機能の値については表 26-2 を参照。level は、記録する状態の重要度や優先順位を示す。優先レベルについては表 26-3 を参照

action

動作フィールドは、メッセージが転送される場所を示す

次は、デフォルトの /etc/syslog.conf ファイルの例です。

```
user.err /dev/sysmsg
user.err /var/adm/messages
user.alert `root, operator`
user.emerg *
```

この例は、次のユーザーメッセージが自動的に記録されることを意味します。

- ユーザーエラーはコンソールに出力され、/var/adm/messages ファイルにも記録されます。
- 早急な対応が必要なユーザーメッセージ (alert) は、root ユーザーと operator ユーザーに送信されます。
- ユーザー緊急メッセージは、各ユーザーに送信されます。

最も一般的なエラー状態の送信元を表 26-2 に示します。最も一般的な優先順位を、重要度順に表 26-3 に示します。

表 26-2 syslog.conf メッセージの機能

送信元	説明
kern	カーネル
auth	認証
daemon	すべてのデーモン
mail	メールシステム
lp	スプールシステム
user	ユーザープロセス

注 - /etc/syslog.conf ファイルで有効化できる syslog 機能の数に制限はありません。

表 26-3 syslog.conf メッセージの優先レベル

優先順位	説明
emerg	システムの緊急事態
alert	すぐに修正が必要なエラー
crit	致命的なエラー
err	その他のエラー
info	情報メッセージ
debug	デバッグ用の出力
none	この設定は出力を記録しない

▼ システムのメッセージ記録をカスタマイズする方法

1. スーパーユーザーになります。
2. /etc/syslog.conf ファイルを編集します。syslog.conf(4) のマニュアルページで説明している構文に従って、メッセージの送信元、優先順位、およびメッセージの格納場所を追加または変更します。
3. 変更を保存して編集を終了します。

例 — システムメッセージ記録をカスタマイズする

次の `/etc/syslog.conf` の `user.emerg` 機能の例は、ユーザー緊急メッセージを `root` ユーザーと個別のユーザーに送信します。

```
user.emerg                                `root, *'
```

リモートコンソールメッセージングを有効にする

次の新しいリモートコンソール機能を使うと、リモートシステムの問題を解決しやすくなります。

- `consadm` コマンドでは、補助 (またはリモート) コンソールとしてシリアルデバイスを選択できます。`consadm` コマンドを使用して、システム管理者は1つまたは複数のシリアルポートを構成して、出力先が変更されたコンソールメッセージを表示したり、システムの実行レベルが変わったときに `sulogin` セッションをサポートしたりできます。この機能を使用して、モデム付きのシリアルポートにダイヤルインしてコンソールメッセージを監視し、`init` 状態の変更を表示できます (詳細については、`sulogin(1M)` のマニュアルページと以下の詳しい手順を参照)。

補助コンソールとして構成されたポートからシステムにログインすることもできますが、このポートは主に、デフォルトコンソールに表示される情報を表示する出力デバイスです。ブートスクリプトやその他のアプリケーションがデフォルトコンソールに対して読み書きを行う場合、書き込み出力はすべての補助コンソールに出力されますが、入力にはデフォルトコンソールからだけ読み込まれます (対話型ログインセッションでの `consadm` コマンドの使用方法については、372 ページの「対話型ログインセッション中に `consadm` コマンドを使用する」を参照)。

- コンソール出力はカーネルメッセージと `syslog` メッセージからなり、新しい仮想デバイス `/dev/sysmsg` に書き込まれます。さらに、`rc` スクリプト起動メッセージが `/dev/msglog` に書き込まれます。以前のリリースでは、これらのメッセージはすべて `/dev/console` に書き込まれていました。

スクリプトメッセージを補助コンソールに表示したい場合は、コンソール出力を `/dev/console` に出力しているスクリプトで出力先を `/dev/msglog` に変更する必要があります。メッセージ出力を補助デバイスに出力変更したい場合は、`/dev/console` を参照しているプログラムで `syslog()` または `strlog()` を使用するように明示的に変更してください。

- `consadm` コマンドは、デーモンを実行して補助コンソールデバイスを監視します。補助コンソールに指定された表示デバイスがハングアップしたりキャリア信号がなくなって切り離されると、そのデバイスは補助コンソールデバイスのリストから削除され、アクティブでなくなります。1つまたは複数の補助コンソールを有効にしても、メッセージがデフォルトコンソールに表示されなくなるわけではありません。メッセージは引き続き `/dev/console` に出力されます。

実行レベルの変更中に補助コンソールメッセージングを使用する

実行レベルの変更中に補助コンソールメッセージングを使う場合は、次の点に注意してください。

- システムのブート時に実行する rc スクリプトにユーザーの入力がある場合は、補助コンソールから入力を行うことはできません。入力はデフォルトコンソールから行う必要があります。
- 実行レベルの変更中に、スーパーユーザーパスワード入力を要求するために `sulogin` プログラムが `init` によって呼び出されます。このプログラムは、デフォルトのコンソールデバイスだけでなく各補助デバイスにもスーパーユーザーパスワードの入力要求を送信するように変更されています。
- システムがシングルユーザーモードで動作し、1つまたは複数の補助コンソールが `consadm` コマンドによって有効になっていると、最初のデバイスでコンソールログインセッションが実行され、正確なスーパーユーザーパスワードを要求する `sulogin` プロンプトが表示されます。コンソールデバイスから正しいパスワードを受け取ると、`sulogin` は他のすべてのコンソールデバイスからの入力を受信できないようにします。
- コンソールの1つがシングルユーザー特権を取得すると、デフォルトコンソールとその他の補助コンソールにメッセージが出力されます。このメッセージは、どのデバイスから正しいスーパーユーザーパスワードが入力され、コンソールになったかを示します。シングルユーザーシェルが動作する補助コンソールのキャリア信号が失われると、次のどちらかのアクションが起ることがあります。
 - 補助コンソールが実行レベル 1 のシステムを表している場合は、システムはデフォルトの実行レベルに移行します。
 - 補助コンソールが実行レベル S のシステムを表している場合は、シェルから `init s` または `shutdown` コマンドが入力されたデバイスに「ENTER RUN LEVEL (0-6, s or S):」というメッセージが表示されます。このデバイスのキャリア信号も失われている場合は、キャリア信号を復活して正確な実行レベルを入力する必要があります。`init` や `shutdown` コマンドを実行しても、実行レベルプロンプトが再表示されることはありません。
- シリアルポートを使用してシステムにログインしている場合には、`init` または `shutdown` コマンドを使用して別の実行レベルに移行すると、このデバイスが補助コンソールかどうかに関係なくログインセッションは失われます。この状況は、補助コンソール機能がない Solaris リリースと同じです。
- `consadm` コマンドを使って補助コンソールにするデバイスを選択すると、システムをリブートするか補助コンソールの選択を解除するまで、そのデバイスは補助コンソールとして有効です。ただし、`consadm` コマンドには、複数のシステムリブートにまたがってデバイスを補助コンソールとして使用するオプションがありません (以下の詳しい手順を参照)。

対話型ログインセッション中に consadm コマンドを使用する

シリアルポートに接続されている端末からシステムにログインしてから consadm コマンドを使ってこの端末にコンソールメッセージを表示して対話型ログインセッションを行う場合、次の点に注意してください。

- この端末で対話型ログインセッションを行う場合、補助コンソールがアクティブだと、コンソールメッセージは /dev/sysmsg デバイスまたは /dev/msglog デバイスに送られます。
- この端末からコマンドを発行すると、入力はデフォルトコンソール (/dev/console) ではなく対話型セッションに送られます。
- init コマンドを実行して実行レベルを変更すると、リモートコンソールソフトウェアは対話型セッションを終了し、sulogin プログラムを実行します。この時点では、入力はこの端末からだけ可能で、入力はコンソールデバイスから行われたかのように扱われます。そのため、371 ページの「実行レベルの変更中に補助コンソールメッセージングを使用する」の説明のとおり、sulogin プログラムにパスワードを入力できます。

次に、(補助) 端末から正しいパスワードを入力すると、補助コンソールは、対話型 sulogin セッションを実行し、デフォルトコンソールおよび競合する補助コンソールを使えなくします。つまり、その端末は実質的にシステムコンソールとして機能します。

- この端末から実行レベル 3 または別の実行レベルに変更できます。実行レベルを変更すると、すべてのコンソールデバイスで sulogin が再び実行されます。終了したり、システムが実行レベル 3 で起動されるように指定すると、どの補助コンソールからも入力を行えなくなります。すべての補助コンソールはコンソールメッセージを表示するだけのデバイスに戻ります。

システムが起動する際には、デフォルトのコンソールデバイスから rc スクリプトに情報を入力する必要があります。システムが再び起動すると login プログラムがシリアルポートで実行されるため、別の対話型セッションを開始できます。そのデバイスを補助コンソールに指定していれば、コンソールメッセージはその端末に引き続き出力されます。ただし、端末からの入力はすべて対話型セッションに送られます。

▼ 補助 (リモート) コンソールを有効にする方法

consadm デーモンは、consadm コマンドで補助コンソールを追加するまでポートの監視を開始しません。セキュリティ機能として、コンソールメッセージは、キャリア信号が失われるまでか、補助コンソールデバイスの選択が解除されるまでの間だけ出力変更されます。そのため、consadm コマンドを使うには、そのポートでキャリア信号が確立されている必要があります。

補助コンソールを有効にする方法の詳細については、consadm(1M) のマニュアルページを参照してください。

1. システムにスーパーユーザーとしてログインします。
2. 補助コンソールを有効にします。

```
# consadm -a devicename
```
3. 現在の接続が補助コンソールであることを確認します。

```
# consadm
```

例 — 補助 (リモート) コンソールを有効にする

```
# consadm -a /dev/term/a  
# consadm  
/dev/term/a
```

▼ 補助コンソールのリストを表示する方法

1. システムにスーパーユーザーとしてログインします。
2. 次のどちらかの手順に従います。
 - a. 補助コンソールのリストを表示します。

```
# consadm  
/dev/term/a
```
 - b. 持続的補助コンソールのリストを表示します。

```
# consadm -p  
/dev/term/b
```

▼ システムリブート後も補助 (リモート) コンソールを有効にする方法

1. システムにスーパーユーザーとしてログインします。
2. 複数のシステムリブート後も補助コンソールを有効にします。

```
# consadm -a -p devicename
```

このデバイスが持続的な補助コンソールのリストに追加されます。
3. デバイスが持続的な補助コンソールのリストに追加されているか確認します。

```
# consadm
```

例 — システムリブート後も補助 (リモート) コンソールを有効にする

```
# consadm -a -p /dev/term/a
# consadm
/dev/term/a
```

▼ 補助 (リモート) コンソールを無効にする方法

1. システムにスーパーユーザーとしてログインします。
2. 次のどちらかの手順に従います。
 - a. 補助コンソールを無効にします。

```
# consadm -d devicename
```

または
 - b. 補助コンソールを無効にし、持続的な補助コンソールのリストから削除します。

```
# consadm -p -d devicename
```
3. 補助コンソールが無効になっていることを確認します。

```
# consadm
```

例 — 補助 (リモート) コンソールが無効になっていることを確認する

```
# consadm -d /dev/term/a
# consadm
```

第 27 章

コアファイルの管理 (手順)

この章では、`coreadm` コマンドを使って、コマンドファイルを管理する方法について説明します。

コアファイルの管理に関連する手順の詳細については、375 ページの「コアファイルの管理 (作業マップ)」を参照してください。

コアファイルの管理 (作業マップ)

作業	説明	参照先
1. 現在のコアダンプ構成を表示する	<code>coreadm</code> コマンドを使用して、現在のコアダンプ構成を変更する	378 ページの「現在のコアダンプ構成を表示する方法」
2. コアダンプ構成を変更する	次のいずれかの手順を実行して、コアダンプ構成を変更する	
	コアファイル名パターンを設定する	379 ページの「コアファイル名パターンを設定する方法」
	プロセス別コアファイルのパスを有効にする	379 ページの「プロセス別コアファイルパスを有効にする方法」
	グローバルのコアファイルのパスを有効にする	379 ページの「グローバルのコアファイルパスを有効にする方法」

作業	説明	参照先
3. コアダンプファイルを調べる	proc ツールを使用して、コアダンプファイルを表示する	380 ページの「コアファイルの検査」

コアファイルの管理の概要

コアファイルは、プロセスまたはアプリケーションが異常終了したときに生成されます。コアファイルは `coreadm` コマンドで管理します。

たとえば、`coreadm` コマンドを使用して、プロセスコアファイルをすべて同じシステムディレクトリに置くようにシステムを構成できます。Solaris のプロセスやデーモンが異常終了した場合に、特定のディレクトリにあるコアファイルを調べればよいため、問題の追跡が容易になります。

構成可能なコアファイルのパス

次の 2 つの構成可能な新しいコアファイルのパスは、個別に有効または無効にすることができます。

- プロセス別コアファイルのパスにはデフォルトで `core` が使用されます。このパスはデフォルトで有効になっています。プロセス別コアファイルのパスが有効になっていると、プロセスが異常終了したときに `core` ファイルが生成されます。プロセス別のパスは、親プロセスから新しいプロセスに継承されます。

プロセス別コアファイルは生成されるとプロセスの所有者によって所有され、所有者には読み取り／書き込み権が与えられます。所有者だけがこのファイルを表示できます。

- グローバルコアファイルのパスにはデフォルトで `core` が使用されます。このパスはデフォルトで無効になっています。このパスが有効になっていると、プロセス別コアファイルのパスと同じ内容のコアファイルがグローバルコアファイルのパスに追加で作成されます。

グローバルコアファイルは生成されるとスーパーユーザーによって所有され、スーパーユーザーだけに読み取り／書き込み権が与えられます。アクセス権のないユーザーはこのファイルを表示できません。

プロセスが異常終了すると、以前の Solaris リリースと同じように `core` ファイルが現在のディレクトリに作成されます。しかし、たとえば、グローバルコアファイルのパスが有効で `/corefiles/core` に設定されていると、プロセスが終了するたびにコアファイルが 2 つ、1 つは現在の作業ディレクトリに、1 つは `/corefiles` ディレクトリにそれぞれ作成されます。

デフォルトでは、`setuid` プロセスは、グローバルの設定やプロセス別のパスを使ってコアファイルを生成することはありません。

拡張されたコアファイル名

グローバルコアファイルディレクトリが有効な場合、次の表に示す変数を使って core ファイルを相互に区別できます。

変数名	変数の定義
%p	プロセス ID
%u	実効ユーザー ID
%g	実効グループ ID
%f	実行可能ファイル名
%n	システムノード名。uname -n の出力と同じ
%m	マシン名。uname -m での出力と同じ
%t	time (2) システム呼び出しの 10 進数
%%	リテラル %

たとえば、グローバルコアファイルパスが次のように設定されている場合、

```
/var/core/core.%.f.%.p
```

PID 12345 の sendmail プロセスが異常終了すると、次の core ファイルが作成されます。

```
/var/core/core.sendmail.12345
```

コアファイル名パターンの設定

コアファイル名パターンは、グローバルに設定したりプロセス単位で設定したりできます。さらに、この設定をシステムリブート後も有効になるように保存するかどうかを指定できます。

たとえば、次の coreadm コマンドでは、init プロセスで起動されたすべてのプロセスに対しグローバルのコアファイルパターンを設定します。このパターンは複数のシステムリブート後も有効です。

```
$ coreadm -i /var/core/core.%.f.%.p
```

グローバルのコア値は /etc/coreadm.conf ファイルに格納されるため、この設定値はシステムリブート後も有効になるように保存されます。

次の coreadm コマンドでは、すべてのプロセスに対しプロセス別コアファイル名パターンを設定します。

```
$ coreadm -p /var/core/core.%.f.%.p $$
```

\$\$ 記号には、現在実行中のシェルのプロセス ID を指定します。プロセス別コアファイル名パターンは、すべての子プロセスに継承されます。

グローバルまたはプロセス別のコアファイル名パターンを設定したら、これを `coreadm -e` コマンドで有効にする必要があります。詳細については次の手順を参照してください。

このコマンドをユーザーの `$HOME/.profile` または `.login` ファイルに入れておけば、ユーザーのログインセッションで実行するすべてのプロセスに対しコアファイル名パターンを設定できます。

setuid プログラムがコアファイルを作成できるようにする

`coreadm` コマンドを使って `setuid` プログラムを有効または無効にすれば、次の設定を行うことによって、すべてのシステムプロセスに対して、または各プロセスに対してコアファイルを作成できます。

- グローバル `setuid` オプションが有効になっていると、グローバルコアファイルパスに従って、システムのすべての `setuid` プログラムが `core` ファイルを作成します。
- プロセス別 `setuid` オプションが有効になっていると、プロセス別コアファイルパスに従って、特定の `setuid` プロセスが `core` ファイルを作成します。

デフォルトでは、両方のフラグが無効になっています。セキュリティ上の理由により、グローバルコアファイルパスは、`/` で始まるフルパス名であることが必要です。スーパーユーザーがプロセス別コアファイルを無効にすると、個別のユーザーがコアファイルを得ることはできなくなります。

`setuid` コアファイルはスーパーユーザーによって所有され、スーパーユーザーだけに読み取り/書き込み権が与えられます。通常ユーザーは、たとえ `setuid` コアファイルを生成したプロセスを所有していても、それらのファイルにアクセスできません。

詳細については、`coreadm(1M)` のマニュアルページを参照してください。

現在のコアダンプ構成を表示する方法

現在のコアダンプ構成を表示するには、オプションを指定しないで `coreadm` コマンドを実行します。

```
$ coreadm
      global core file pattern: /var/core/core.%f.%p
      init core file pattern: core
      global core dumps: enabled
      per-process core dumps: enabled
```

```
global setid core dumps: enabled
per-process setid core dumps: disabled
global core dump logging: disabled
```

コアファイル名パターンを設定する方法

1. プロセス別コアファイルを設定するのか、グローバルコアファイルを設定するのかを決めて、次のどちらかの手順に従います。

- a. プロセス別コアファイル名パターンを設定します。

```
$ coreadm -p $HOME/corefiles/%f.%p $$
```

- b. グローバルコアファイル名パターンを設定します。
まずスーパーユーザーになる必要があります。

```
# coreadm -g /var/corefiles/%f.%p
```

▼ プロセス別コアファイルパスを有効にする方法

1. スーパーユーザーになります。
2. プロセス別コアファイルパスを有効にします。

```
# coreadm -e process
```

3. 現在のプロセスのコアファイルパスを表示して構成を確認します。

```
$ coreadm $$
1180: /home/kryten/corefiles/%f.%p
```

▼ グローバルのコアファイルパスを有効にする方法

1. スーパーユーザーになります。
2. グローバルのコアファイルパスを有効にします。

```
# coreadm -e global -g /var/core/core.%f.%p
```

3. 現在のプロセスのコアファイルパスを表示して構成を確認します。

```
# coreadm
global core file pattern: /var/core/core.%f.%p
init core file pattern: core
global core dumps: enabled
per-process core dumps: enabled
global setid core dumps: disabled
```

```
per-process setid core dumps: disabled
global core dump logging: disabled
```

コアファイルの問題解決

エラーメッセージ

```
NOTICE: 'set allow_setid_core = 1' in /etc/system is obsolete
NOTICE: Use the coreadm command instead of 'allow_setid_core'
```

エラーの発生原因

setuid コアファイルを許容する古いパラメータが `/etc/system` ファイルにあります。

解決方法

`/etc/system` ファイルから `allow_setid_core=1` を削除し、`coreadm` コマンドを使ってグローバル setuid コアファイルの設定を有効にします。

コアファイルの検査

一部の `proc` ツールが拡張されてプロセスのコアファイルやライブプロセスが調べられるようになりました。 `proc` ツールは、`/proc` ファイルシステムの機能を操作するユーティリティです。

現在、コアファイルを処理できるツールは `/usr/proc/bin` ディレクトリにある `pstack`、`pmap`、`pldd`、`pflags`、`pcred` です。これらのツールを使用するには、プロセス ID を指定するように、コアファイルの名前をコマンド行に指定します。

`proc` ツールを使用してコアファイルを調べる方法については、`proc(1)` のマニュアルページを参照してください。

例 — `proc` ツールを使用したコアファイルの検査

```
$ ./a.out
Segmentation Fault (coredump)
$ /usr/proc/bin/pstack ./core
core './core' of 19305: ./a.out
000108c4 main      (1, ffbef5cc, ffbef5d4, 20800, 0, 0) + 1c
00010880 _start    (0, 0, 0, 0, 0, 0) + b8
```

第 28 章

システムクラッシュ情報の管理 (手順)

この章では、Solaris 環境でシステムクラッシュ情報を管理する方法を説明します。

システムクラッシュ情報の管理に関連する手順については、381 ページの「システムクラッシュ情報 (作業マップ)」を参照してください。

システムクラッシュ情報 (作業マップ)

次の作業マップは、システムクラッシュ情報の管理に必要な手順を示します。

作業	説明	参照先
1. 現在のクラッシュダンプ構成を表示する	dumpadm コマンドを使用して、現在のクラッシュダンプ構成を表示する	385 ページの「現在のクラッシュダンプ構成を表示する方法」
2. クラッシュダンプ構成を変更する	dumpadm コマンドを使用して、ダンプするデータの種類、システムが専用のダンプデバイスを使用するかどうか、クラッシュダンプファイルを保存するディレクトリ、およびクラッシュダンプファイルが書き込まれた後に残っていなければならない容量を指定する	385 ページの「クラッシュダンプ構成を変更する方法」
3. クラッシュダンプファイルを調べる	mdb コマンドを使用して、クラッシュダンプファイルを表示する	387 ページの「クラッシュダンプを検査する方法」

作業	説明	参照先
4. (省略可能) クラッシュダンプディレクトリが一杯になった場合に復元する	システムがクラッシュした際にメモリーイメージを格納する十分な空き容量が <code>savecore</code> ディレクトリになくても、一部の重要なシステムクラッシュダンプ情報を保存したい場合	387 ページの「クラッシュダンプディレクトリが一杯になった場合に復元する方法 (省略可能)」
5. (省略可能) クラッシュダンプファイルの保存を有効または無効にする	<code>dumpadm</code> コマンドを使用して、クラッシュダンプファイルの保存を有効または無効にする。デフォルトでは、クラッシュダンプファイルは保存される	388 ページの「クラッシュダンプの保存を無効または有効にする方法」

システムクラッシュ (概要)

ハードウェアの障害、入出力の問題、ソフトウェアエラーなどが原因でシステムがクラッシュすることがあります。システムがクラッシュすると、システムはエラーメッセージをコンソールに表示し、物理メモリーのコピーをダンプデバイスに書き込みます。その後、システムは自動的にリブートします。システムがリブートすると、`savecore` コマンドが実行され、ダンプデバイスのデータを取り出して保存されたクラッシュダンプを `savecore` ディレクトリに書き込みます。このクラッシュダンプファイルは、サポートプロバイダにとって、問題を診断する上で貴重な情報となります。

システムクラッシュダンプファイル

システムクラッシュの後で自動的に実行される `savecore` コマンドは、ダンプデバイスからクラッシュダンプ情報を取り出し、`unix.X` と `vmcore.X` という 1 対のファイルを作成します。X はダンプの通し番号です。これらのファイルは 2 つで、保存されたシステムクラッシュダンプの情報を表します。

クラッシュダンプファイルはコアファイルと混同されることがあります。コアファイルは、アプリケーションが異常終了したときに書き込まれるユーザーアプリケーションのイメージです。

クラッシュダンプファイルは、あらかじめ決められたディレクトリに保存されます。これはデフォルトでは `/var/crash/hostname` です。以前の Solaris リリースでは、システムを手動で有効にして物理メモリーのイメージをクラッシュダンプファイルに保存しない限り、システムがリブートされた時にクラッシュダンプファイルが上書きされていました。このリリースでは、クラッシュダンプファイルの保存がデフォルトで有効です。

システムクラッシュ情報は `dumpadm` コマンドで管理します。詳細は、383 ページの「`dumpadm` コマンド」を参照してください。

クラッシュダンプの保存

制御構造体、アクティブなテーブル、動作中またはクラッシュしたシステムカーネルのメモリのイメージなど、カーネルの動作状況についての情報を調べるには、`mdb` ユーティリティを使用します。`mdb` を完全に使いこなすには、カーネルについての詳細な知識が必要ですが、このマニュアルでは説明を省きます。このユーティリティについては、`mdb` (1) のマニュアルページを参照してください。

`savecore` で保存したクラッシュダンプを購入先に送って、システムがクラッシュした原因を解析してもらうことも可能です。

`dumpadm` コマンド

`dumpadm` コマンドを使用して、Solaris 環境でシステムクラッシュダンプ情報を管理します。

- オペレーティングシステムのクラッシュダンプを構成することもできます。`dumpadm` 構成パラメータでは、ダンプ内容、ダンプデバイス、クラッシュダンプファイルが保存されるディレクトリなどを指定します。
- ダンプデータは、圧縮した形式でダンプデバイスに格納されます。カーネルのクラッシュダンプイメージは 4G バイトを超える場合があります。データを圧縮することにより、ダンプが速くなり、ダンプデバイスのディスク領域も少なくて済みます。
- スワップ領域ではなく、専用のダンプデバイスがダンプ構成の一部にあると、クラッシュダンプファイルの保存はバックグラウンドで行われます。つまり、システムを起動する際、`savecore` コマンドが完了するのを待たなくても、次の段階に進むことができます。大容量のメモリーを搭載したシステムでは、`savecore` コマンドが完了する前にシステムが使用可能になります。
- `savecore` コマンドで生成されるシステムクラッシュダンプファイルは、デフォルトで保存されます。
- `savecore -L` コマンドは、動作中の Solaris オペレーティング環境でクラッシュダンプを取得できる新しい機能です。たとえば、パフォーマンスに問題が発生しているときやサービスが停止しているときなどにメモリーのスナップショットをとって、実行中のシステムの問題を解決するのに使用します。システムが実行中で、一部のコマンドがまだ使用できる場合は、`savecore -L` コマンドを使用してシステムのスナップショットをダンプデバイスに保存し、クラッシュダンプファイルをただちに `savecore` ディレクトリに書き込むことができます。システムが実行中であるため、専用のダンプデバイスを構成してある場合のみ、`savecore -L` コマンドを使用できます。

次の表で `dumpadm` の構成パラメータを説明します。

ダンプパラメータ	説明
ダンプデバイス	システムがクラッシュしたときにダンプデータを一時的に保存するデバイス。ダンプデバイスがスワップ領域でない場合は、 <code>savecore</code> がバックグラウンドで実行されるため、ブートプロセスの速度が上がる
<code>savecore</code> ディレクトリ	システムのクラッシュダンプファイルを保存するディレクトリ
ダンプ内容	ダンプするメモリーデータの種類
最小空き容量	クラッシュダンプファイルを保存した後で <code>savecore</code> ディレクトリに必要な最小空き容量。空き容量を指定しないと、デフォルトで 1M バイトになる

詳細については、`dumpadm(1M)` のマニュアルページを参照してください。

`dumpadm` コマンドで管理するダンプ構成パラメータは、`/etc/dumpadm.conf` ファイルに保存されます。

注 - `/etc/dumpadm.conf` ファイルは、手作業で編集しないでください。このファイルを手作業で編集すると、システムダンプ構成の整合性が失われる恐れがあります。

dumpadm コマンドの動作

`dumpadm` コマンドは、システム起動時に `/etc/init.d/savecore` スクリプトによって呼び出され、`/etc/dumpadm.conf` ファイルの情報に基づいてクラッシュダンプパラメータの構成を行います。

`dumpadm` コマンドは、`/dev/dump` インタフェースを通してダンプデバイスとダンプ内容を初期化します。

ダンプ構成が完了すると、`savecore` スクリプトは、`/etc/dumpadm.conf` ファイルの内容を解析してクラッシュダンプファイルのディレクトリの場所を探します。次に、`savecore` を呼び出して、クラッシュダンプがあるかどうかを調べたり、クラッシュダンプディレクトリにある `minfree` ファイルの内容を確認したりします。

ダンプデバイスとボリュームマネージャ

可用性とパフォーマンス上の理由のため、Solaris ボリュームマネージャーで管理されている専用ダンプデバイスを構成しないでください。スワップ領域を Solaris ボリュームマネージャーの管理下に置くことはできますが (この方法を推奨します)、ダンプデバイスは別に確保してください。

システムクラッシュダンプ情報の管理

システムクラッシュ情報を処理する場合には、次の点に注意してください。

- システムクラッシュ情報にアクセスして管理するには、スーパーユーザーでログインする必要があります。
- システムクラッシュダンプを保存するオプションを無効にしないでください。システムクラッシュファイルにより、システムクラッシュの原因を判断する非常に有効な方法が提供されます。
- また、重要なシステムクラッシュ情報は、カスタマサービス担当者へ送信するまでは削除しないでください。

▼ 現在のクラッシュダンプ構成を表示する方法

1. スーパーユーザーになります。
2. 現在のクラッシュダンプ構成を表示します。

```
# dumpadm          Dump content: kernel pages
                   Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/venus
Savecore enabled: yes
```

上記の出力例の意味は次のとおりです。

- ダンプの内容は、カーネルメモリーページである
- カーネルメモリーがスワップデバイス /dev/dsk/c0t3d0s1 にダンプされる。
swap -l コマンドにより、すべてのスワップ領域を識別できる
- システムクラッシュダンプファイルは /var/crash/venus ディレクトリに保存される
- システムクラッシュダンプファイルの保存は有効に設定されている

▼ クラッシュダンプ構成を変更する方法

1. スーパーユーザーになります。
2. 現在のクラッシュダンプ構成を確認します。

```
# dumpadm
Dump content: kernel pages
Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/pluto
Savecore enabled: yes
```

上記の出力は、Solaris 9 リリースを実行するシステムのデフォルトダンプ構成を示しています。

3. クラッシュダンプ構成を変更します。

```
# dumpadm -c content -d dump-device -m nnnk | nnnm | nnn% -n -s savecore-dir
```

-c content	ダンプするデータの種類を指定する。すべてのカーネルメモリーをダンプするには kernel を、すべてのメモリーをダンプするには all を、カーネルメモリーとクラッシュ時に実行中だったスレッドを持つプロセスのメモリーページとをダンプするには curproc を使用する。デフォルトはカーネルメモリー
-d dump-device	システムがクラッシュしたときに、ダンプデータを一時的に保存するデバイスを指定する。デフォルトのダンプデバイスは 1 次スワップデバイス
-m nnnk nnnm nnn%	現在の savecore ディレクトリに minfree ファイルを作成することにより、クラッシュダンプファイルを保存する最小限の空き容量を指定する。このパラメータは K バイト (nnnk)、M バイト (nnnm)、またはファイルシステムサイズのパーセント (nnn%) で指定できる。savecore コマンドは、クラッシュダンプファイルを書き込む前にこのファイルを調べる。クラッシュダンプファイルを書き込むと空き容量が minfree の値より少なくなる場合、ダンプファイルは書き込まれず、エラーメッセージが記録される。このような問題を解決するには、387 ページの「クラッシュダンプディレクトリが一杯になった場合に復元する方法 (省略可能)」を参照
-n	システムがリブートするときに、savecore を実行しないように指定する。このダンプ構成は推奨できない。システムクラッシュ情報がスワップデバイスに書き込まれているときに、savecore が実行されないと、クラッシュダンプ情報はシステムがスワップを開始すると上書きされる
-s	クラッシュダンプファイルを保存する別のディレクトリを指定する。デフォルトのディレクトリは /var/crash/hostname で、hostname は uname -n コマンドの出力

例 —クラッシュダンプ構成を変更する

次の例は、すべてのメモリーを専用のダンプデバイス /dev/dsk/c0t1d0s1 にダンプします。また、クラッシュダンプファイルを保存した後に残っていなければならない最小空き容量は、ファイルシステム容量の 10% です。

```
# dumpadm
  Dump content: kernel pages
  Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/pluto
  Savecore enabled: yes
# dumpadm -c all -d /dev/dsk/c0t1d0s1 -m 10%
  Dump content: all pages
  Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
  Savecore enabled: yes
```

▼ クラッシュダンプを検査する方法

1. スーパーユーザーになります。
2. mdb ユーティリティを使用して、クラッシュダンプを検査します。

```
# /usr/bin/mdb [-k] crashdump-file
```

`-k` オペレーティングシステムのクラッシュダンプファイルの場合のカーネルデバッグモードを指定します。

`crashdump-file` オペレーティングシステムのクラッシュダンプファイルを指定します。

3. クラッシュ状態情報を表示します。

```
# /usr/bin/mdb file-name
> ::status
.
.
.
> ::system
.
.
.
```

例 — クラッシュダンプを検査する

次の例は、mdb ユーティリティからのサンプル出力を示します。このシステムのシステム情報と `/etc/system` ファイルに設定される調整可能パラメータが表示されます。

```
# /usr/bin/mdb -k unix.0
Loading modules: [ unix krtld genunix ip nfs ipc ptm ]
> ::status
debugging crash dump /dev/mem (64-bit) from ozlo
operating system: 5.9 Generic (sun4u)
> ::system
set ufs_ninode=0x9c40 [0t40000]
set ncsiz=0x4e20 [0t20000]
set pt_cnt=0x400 [0t1024]
```

▼ クラッシュダンプディレクトリが一杯になった場合に復元する方法 (省略可能)

ここでは、システムがクラッシュしてもメモリーイメージを格納する十分な空き容量が `savecore` ディレクトリにないが、それでも、一部の重要なシステムクラッシュダンプ情報を保存したい場合を考えます。

1. システムがリブートした後で、スーパーユーザーとしてログインします。
2. すでにサービスプロバイダに送ってある既存のクラッシュダンプファイルを削除して、**savecore** ディレクトリ (通常は `/var/crash/hostname`) を整理します。あるいは、**savecore** コマンドを実行し、十分な容量を持つ別のディレクトリを指定します (次の手順を参照してください)。
3. 手作業で **savecore** コマンドを実行し、必要なら別の **savecore** ディレクトリを指定します。

```
# savecore [ directory ]
```

▼ クラッシュダンプの保存を無効または有効にする方法

1. スーパーユーザーになります。
2. システム上のクラッシュダンプの保存を有効または無効にします。

```
# dumpadm -n | -y
```

例 — クラッシュダンプの保存を無効にする

次の例は、システムでのクラッシュダンプの保存を無効にします。

```
# dumpadm -n
Dump content: all pages
Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
Savecore enabled: no
```

例 — クラッシュダンプの保存を有効にする

次の例は、システムでのクラッシュダンプの保存を有効にします。

```
# dumpadm -y
Dump content: all pages
Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
Savecore enabled: yes
```

第 29 章

ソフトウェアで発生するさまざまな問題の解決 (手順)

この章では、ときどき発生するが比較的修正しやすい、さまざまなソフトウェアの問題について説明します。特定のソフトウェアアプリケーションや内容に関連しない問題 (リブートの失敗やファイルシステムがフルになるなど) の解決方法も含まれます。これらの問題の解決方法は、この後の節で説明します。

この章で説明する情報は次のとおりです。

- 389 ページの「リブートが失敗した場合の対処」
- 393 ページの「システムがハングした場合の対処」
- 394 ページの「ファイルシステムが一杯になった場合の対処」
- 395 ページの「コピーまたは復元後にファイルの ACL が消失した場合の対処」
- 396 ページの「バックアップ時の問題の解決」

リブートが失敗した場合の対処

システムがリブートに失敗した場合またはリブートしたがクラッシュした場合は、システムのブートを妨害しているソフトウェアまたはハードウェアの障害があると考えられます。

システムがブートしない原因	問題の解決方法
システムが <code>/platform/`uname -m`/kernel/unix</code> を見つけられない	SPARC システムの PROM 内の <code>boot-device</code> 設定を変更しなければならない。デフォルトのブートシステムの変更については、『Solaris のシステム管理 (基本編)』の「SPARC: システムのブート (手順)」を参照してください。

システムがブートしない原因	問題の解決方法
IA システムで、デフォルトのブートデバイスが存在しない。「Not a UFS filesystem.」というメッセージが表示される	Configuration Assistant/Boot (構成用補助) フロッピーディスクを使用してシステムをブートし、ブートするディスクを選択する
/etc/passwd ファイル内に無効なエントリが存在する	無効な passwd ファイルから復元する方法については、『Solaris のシステム管理 (基本編)』の「SPARC: システムのブート (手順)」または『Solaris のシステム管理 (基本編)』の「IA: システムのブート (手順)」を参照してください。
ディスクなどのデバイスに、ハードウェアの問題がある	ハードウェアの接続を確認する <ul style="list-style-type: none"> ■ 装置が接続されていることを確認する ■ すべてのスイッチが適切に設定されていることを確認する ■ すべてのコネクタおよびケーブル (Ethernet ケーブルも含む) を検査する ■ すべて異常がなければ、システムの電源を切り、10 秒 ~ 20 秒ほど待って、もう一度電源を投入する

上記のリストで問題が解決できない場合は、ご購入先にお問い合わせください。

SPARC: 64 ビット Solaris のブートで発生する問題の解決

64 ビット Solaris リリースを UltraSPARC システムにインストールすると、次のどの条件も該当しない場合は、64 ビットカーネルが自動的にブートされます。

- 64 ビットカーネルをブートするのに、UltraSPARC システムに FLASH PROM アップグレードが必要な場合があります。UltraSPARC システムにファームウェアのアップグレードが必要かどうかを知るには、ハードウェアメーカーの資料を参照してください。
- Open Boot PROM の boot-file パラメータに kernel/unix が設定されています。64 ビットカーネルのブートができない場合に、このパラメータがそのように設定されているなら設定を解除してシステムをリブートします。
- 64 ビット Solaris のすべての構成要素がシステムに完全にインストールされ、正しいファームウェアがインストールされていても、UltraSPARC システムによっては、デフォルトで 64 ビット Solaris カーネルがブートされない場合があります。64 ビット Solaris カーネルがブートされなければ、64 ビットアプリケーションを実行することはできません。

この問題の詳細や 64 ビット Solaris カーネルのブートをデフォルトで有効にする方法については、boot (1M) のマニュアルページを参照してください。

システムがどの Solaris カーネルを実行しているかを知るには、`isainfo -kv` コマンドが常に使用できます。

```
$ isainfo -kv
64-bit sparcv9 kernel modules
```

この例では、64 ビット Solaris カーネルが実行されています。

32 ビット Solaris システムで 64 ビット Solaris オペレーティング環境をブートすることはできません。

ルートパスワードを忘れた場合の対処

ルートパスワードを忘れると、システムにログインできなくなります。その場合、次の手順を実行する必要があります。

- キーボードの停止シーケンスを使用して、システムを停止する。
- ブートサーバー、インストールサーバー、またはローカル CD-ROM からブートする。
- ルート (/) ファイルシステムをマウントする。
- `/etc/shadow` ファイルからルートパスワードを削除する。
- システムをリブートする。
- ログインして、ルートのパスワードを設定する。

この手順の詳細については、『Solaris のシステム管理 (基本編)』の「SPARC: システムのブート」および『Solaris のシステム管理 (基本編)』の「IA: システムのブート」を参照してください。

次の例では、SPARC システムおよび IA システムでルートパスワードを忘れた場合の対処方法について説明します。

SPARC: 例 — ルートパスワードを忘れた場合の対処方法

次の例では、ルートパスワードを忘れた場合にネットワークからブートして回復する方法について説明します。この例では、ブートサーバーがすでに有効になっているものとします。システムのリブート後に、必ず新しいルートパスワードを適用してください。

```
(キーボードのシステム停止シーケンスを使用します
— Stop-A キーを押してシステムを停止します。)
ok boot net -s
# mount /dev/dsk/c0t3d0s0 /a
# cd /a/etc
```

```
# TERM=vt100
# export TERM
# vi shadow
(Remove root's encrypted password string)
# cd /
# umount /a
# init 6
```

IA: 例 — ルートパスワードを忘れた場合の対処方法

次の例では、ルートパスワードを忘れた場合に、ネットワークからブートして回復する方法について説明します。この例では、ブートサーバーがすでに有効になっているものとします。システムのリブート後に、必ず新しいルートパスワードを適用してください。

(キーボードのシステム停止シーケンスを使用して、システムを停止します。)
Type any key to continue

SunOS Secondary Boot version 3.00

Solaris Intel Platform Edition Booting System

Running Configuration Assistant...

Autobooting from Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a

If the system hardware has changed, or to boot from a different device, interrupt the autoboot process by pressing ESC.

Press ESCape to interrupt autoboot in 5 seconds.

.
. .
. . .

Boot Solaris

Select one of the identified devices to boot the Solaris kernel and choose Continue.

To perform optional features, such as modifying the autoboot and property settings, choose Boot Tasks.

An asterisk (*) indicates the current default boot device.

> To make a selection use the arrow keys, and press Enter to mark it [X].

```
[X] NET : DEC 21142/21143 Fast Ethernet
on Board PCI at Dev 3
[ ] DISK: (*) Target 0, QUANTUM FIREBALL1280A
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] DISK: Target 1:ST5660A
```



```

on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] DISK: Target 0:Maxtor 9 0680D4
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] CD : Target 1:TOSHIBA CD-ROM XM-5602B 1546
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1

F2_Continue F3_Back F4_Boot Tasks F6_Help
.
.
.
<<< Current Boot Parameters>>>
Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a
Boot args: kernel/unix -r

Select the type of installation you want to perform:

1 Solaris Interactive
2 Custom JumpStart
3 Solaris Web Start

Enter the number of your choice followed by <ENTER> the key.

If you enter anything else, or if you wait for 30 seconds,
an interactive installation will be started.

Select type of installation: b -s
.
.
.
# mount /dev/dsk/c0t0d0s0 /a
.
.
.
# cd /a/etc
# vi shadow
(Remove root's encrypted password string)
# cd /
# umount /a
# init 6

```

システムがハングした場合の対処

ソフトウェアプロセスに問題がある場合、システムは完全にクラッシュせずに凍結、つまりハングすることがあります。ハングしたシステムから回復するには、次の手順に従ってください。

1. システムがウィンドウ環境を実行していたかどうかを調べて、次の推奨事項に従ってください。これらのリストで問題が解決できなかった場合は、手順 2 に進みます。

- コマンドを入力しているウィンドウの中に、ポインタがあることを確認します。
 - 間違っ て Control-s キー (画面を凍結する) を押した場合は、Control-q キーを押します。Control-s キーはウィンドウだけを凍結し、画面全体は凍結しません。ウィンドウが凍結している場合は、他のウィンドウを試します。
 - 可能であれば、ネットワーク上の他のシステムからリモートでログインします。pgrep コマンドを使用して、ハングしているプロセスを見つけます。ウィンドウシステムがハングしている場合は、そのプロセスを特定して強制終了します。
2. Control-\ キーを押して、動作しているプログラムを強制終了します。core ファイルが書き出されることがあります。
 3. Control-c キーを押して、動作している可能性があるプログラムに割り込みをかけます。
 4. リモートからログインして、システムをハングさせているプロセスを特定して強制終了します。
 5. リモートからログインしてスーパーユーザーになり、システムをリブートします。
 6. システムがまだ応答しない場合は、強制的にクラッシュダンプしてリブートします。強制的にクラッシュダンプしてブートする方法については、『Solaris のシステム管理 (基本編)』の「SPARC: クラッシュダンプを強制してシステムをリブートする」または『Solaris のシステム管理 (基本編)』の「IA: クラッシュダンプを強制してシステムをリブートする」を参照してください。
 7. システムがまだ応答しない場合は、電源を切ってから数分待ち、もう一度電源を入れます。
 8. システムがまったく応答しない場合は、ご購入先にお問い合わせください。

ファイルシステムが一杯になった場合の対処

ルート (/) ファイルシステムや他のファイルシステムが一杯になると、次のようなメッセージがコンソールウィンドウに表示されます。

```
.... file system full
```

ファイルシステムが一杯になる原因はいくつかあります。次の節では、一杯になったファイルシステムを回復する方法をいくつか説明します。ファイルシステムが一杯にならないように、古い使用されていないファイルを日常的に整理する方法については、第 16 章を参照してください。

大規模ファイルまたはディレクトリを作成したために、ファイルシステムが一杯になる

エラーの原因	問題の解決方法
ファイルかディレクトリを間違った場所にコピーした。これは、アプリケーションがクラッシュして、大きな core ファイルをファイルシステムに書き込んだときにも発生する	スーパーユーザーとしてログインし、特定のファイルシステムで <code>ls -tl</code> コマンドを使用し、新しく作成された大きなファイルを特定して削除する。core ファイルを削除する方法については、238 ページの「core ファイルを見つけて削除する方法」を参照してください。

システムのメモリーが不足したために、tmpfs ファイルシステムが一杯になる

エラーの原因	問題の解決方法
これは、tmpfs に許可されているよりも多く書き込もうとした、または現在のプロセスがメモリーを多く使用している場合に発生する	tmpfs に関連するエラーメッセージから回復する方法については、tmpfs (7FS) のマニュアルページを参照してください。

コピーまたは復元後にファイルの ACL が消失した場合の対処

エラーの原因	問題の解決方法
ACL を持つファイルまたはディレクトリを /tmp ディレクトリにコピーすると、ACL 属性が消失する。/tmp ディレクトリは、通常、一時ファイルシステムとしてマウントされ、ACL などの UFS ファイルシステム属性はサポートしない	代わりに、/var/tmp ディレクトリにファイルをコピーまたは復元する

バックアップ時の問題の解決

この節では、データをバックアップまたは復元するときのいくつかの基本的な問題の解決方法について説明します。

ファイルシステムのバックアップ中に、ルート (/) ファイルシステムが一杯になる

ファイルシステムをバックアップしている際に、ルート (/) ファイルシステムが一杯になる場合があります。このとき、媒体には何も書き込まれず、`ufsdump` コマンドは、媒体の 2 番目のボリュームを挿入するようにプロンプトを表示します。

エラーの原因	問題の解決方法
<code>-f</code> オプションに無効な宛先デバイス名を使用した場合、 <code>ufsdump</code> コマンドはファイルシステムを /dev ディレクトリに書き込み、このファイルシステムが一杯になる。たとえば、 <code>/dev/rmt/0</code> ではなく <code>/dev/rmt/st0</code> と入力した場合、バックアップファイルはテープドライブには送信されず、 <code>/dev/rmt/st0</code> がディスクに作成される	<code>/dev</code> ディレクトリで <code>ls -tl</code> コマンドを使用して、新しく作成された異常に大きなファイルを特定して削除する

バックアップコマンドと復元コマンドが対応していることを確認する

`ufsrestore` コマンドを使用できるのは、`ufsdump` コマンドを使用してバックアップしたファイルを復元する場合だけです。`tar` コマンドを使用してバックアップした場合は、`tar` コマンドを使用して復元します。他のコマンドで書き込まれたテープを `ufsrestore` コマンドを使用して復元しようとした場合、テープが `ufsdump` フォーマットでないことを知らせるエラーメッセージが表示されます。

現在のディレクトリが間違っていないことを確認する

ファイルを復元する場合に、間違った場所に復元してしまうことがよくあります。ufsdump コマンドは、常にファイルシステムのルートからのフルパス名でファイルをコピーします。したがって ufsrestore コマンドを実行する前に、ファイルシステムのルートディレクトリに移動しなければなりません。それよりも下のディレクトリでファイルを復元すると、そのディレクトリの下に完全なファイルツリーが作成されます。

対話型コマンド

対話型コマンドを使用すると、次の例のような ufsrestore> プロンプトが表示されます。

```
# ufsrestore ivf /dev/rmt/0
Verify volume and initialize maps
Media block size is 126
Dump   date: Thu Aug 30 10:13:46 2001
Dumped from: the epoch
Level 0 dump of /export/home on starbug:/dev/dsk/c0t0d0s7
Label: none
Extract directories from tape
Initialize symbol table.
ufsrestore>
```

ufsrestore> プロンプトでは、『Solaris のシステム管理 (基本編)』の「UFS バックアップおよび復元コマンド (参照情報)」にリストされているコマンドを使用して、ファイルの検索、復元するファイルのリストの作成、ファイルの復元を行うことができます。

第 30 章

ファイルアクセスでの問題の解決 (手順)

この章では、適切でないアクセス権と検索パスに関連する問題などのファイルアクセスでの問題を解決する手順について説明します。

この章で説明する手順は次のとおりです。

- 399 ページの「検索パスに関連する問題を解決する (コマンドが見つかりません)」
- 402 ページの「ファイルアクセスの問題を解決する」
- 402 ページの「ネットワークアクセスで発生する問題の把握」

以前は使用できていたプログラム、ファイル、またはディレクトリにアクセスできないため、システム管理者に問い合わせる場合があります。このようなときは、次の 3 点を調べてください。

- ユーザーの検索パスが変更されているか、または検索パス中のディレクトリが適切な順序であるか
- ファイルまたはディレクトリに適切なアクセス権や所有権があるか
- ネットワーク経由でアクセスするシステムの構成が変更されているか

この章では、これらの 3 点を確認する方法を簡単に説明して、可能な解決策を提案します。

検索パスに関連する問題を解決する (コマンドが見つかりません)

「コマンドが見つかりません」というメッセージは、次のいずれかを示しています。

- コマンドがそのシステムに存在しない
- コマンドのディレクトリが検索パスに存在しない

検索パスの問題を解決するには、コマンドが格納されているディレクトリのパス名を知る必要があります。

間違ったバージョンのコマンドが見つかってしまうのは、同じ名前のコマンドを持つディレクトリが検索パスにある場合です。この場合、正しいディレクトリが検索パスの後ろの方にあるか、まったく存在しない可能性があります。

現在の検索パスを表示するには、`echo $PATH` コマンドを使用します。次に入力例を示します。

```
$ echo $PATH
/home/kryten/bin:/sbin:/usr/sbin:/usr/bin:/usr/dt:/usr/dist/exe
```

間違ったバージョンのコマンドを実行しているかどうかを調べるには、`which` コマンドを使用します。次に入力例を示します。

```
$ which acroread
/usr/doctools/bin/acroread
```

注 - `which` コマンドは、`.cshrc` ファイルの中のパス情報を調べます。`.cshrc` ファイルに `which` コマンドの認識する別名を定義している場合に、`Bourne` シェルか `Korn` シェルから `which` コマンドを実行すると、間違った結果が返される場合があります。正しい結果を得るために、`which` コマンドは `C` シェルで使用してください。`Korn` シェルの場合は、`whence` コマンドを使用します。

▼ 検索パスの問題を診断して解決する方法

1. 現在の検索パスを表示して、コマンドが入っているディレクトリがユーザーのパス内に存在しない (あるいはスペルが間違っている) ことを確認します。

```
$ echo $PATH
```

2. 次のことをチェックします。

- 検索パスは正しいか
- 検索パスは、コマンドの他のバージョンが存在する他の検索パスの前に指定されているか
- 検索パスのいずれかにコマンドが存在するか

パスを修正する必要がある場合は、手順 3 に進みます。修正する必要がない場合は、手順 4 に進みます。

3. 次の表に示すように、適切なファイルでパスを追加します。

シェル	ファイル	構文	注
Bourne と Korn	<code>\$HOME/.profile</code>	<pre>\$ PATH=\$HOME/bin:/sbin:/usr/local /bin ... \$ export PATH</pre>	パス名はコロンで区切る
C	<code>\$HOME/.cshrc</code> または <code>\$HOME/.login</code>	<pre>hostname% set path=(~bin/sbin /usr/local/bin ...)</pre>	パス名は空白文字で区切る

4. 次のように、新しいパスを有効にします。

シェル	パスが指定されている ファイル	パスを有効にするコマンド
Bourne と Korn	<code>.profile</code>	<pre>\$. ./profile</pre>
C	<code>.cshrc</code>	<pre>hostname% source .cshrc</pre>
	<code>.login</code>	<pre>hostname% source .login</pre>

5. 新しいパスを確認します。

```
$ which command
```

例 — 検索パスの問題を診断および修正する

この例は、`which` コマンドを使用して、`mytool` の実行可能ファイルが検索パス中のどのディレクトリにも存在しないことを示しています。

```
venus% mytool
mytool: Command not found
venus% which mytool
no mytool in /sbin /usr/sbin /usr/bin /etc /home/ignatz/bin .
venus% echo $PATH
/sbin /usr/sbin /usr/bin /etc /home/ignatz/bin
venus% vi ~/.cshrc
(Add appropriate command directory to the search path)
venus% source .cshrc
venus% mytool
```

コマンドを見つけることができなかつた場合は、マニュアルページでそのディレクトリパスを調べます。たとえば、lpsched コマンド (lp プリンタデーモン) を見つけることができなかつた場合、lpsched(1M) のマニュアルページを調べると、そのパスが /usr/lib/lp/lpsched であることが解かります。

ファイルアクセスの問題を解決する

以前はアクセスできていたファイルまたはディレクトリにアクセスできない場合は、そのファイルまたはディレクトリのアクセス権または所有権が変更されていることがあります。

ファイルとグループの所有権の変更

誰かがスーパーユーザーとしてファイルを編集したために、ファイルやディレクトリの所有権が変更されていることがあります。新しいユーザーのためにホームディレクトリを作成するときは、そのホームディレクトリのドット (.) ファイルの所有者をそのユーザーにしてください。ユーザーが「.」を所有していない場合、そのユーザーは自分のホームディレクトリにファイルを作成できません。

アクセスに関する問題は、グループの所有権が変更されたとき、またはユーザーがメンバーであるグループが /etc/group データベースから削除されたときにも発生します。

アクセスに問題があるファイルのアクセス権や所有権の変更方法については、『Solaris のシステム管理 (セキュリティサービス)』の「ファイルのセキュリティの適用 (手順)」を参照してください。

ネットワークアクセスで発生する問題の把握

リモートコピーコマンド rcp を使用してネットワーク上でファイルをコピーするときに問題が発生した場合、リモートシステム上のディレクトリやファイルは、アクセス権の設定によりアクセスが制限されている可能性があります。他に考えられる問題の原因は、リモートシステムとローカルシステムがアクセスを許可するように構成されていないことです。

ネットワークアクセスで発生する問題と AutoFS 経由でシステムにアクセスする場合に発生する問題については、『Solaris のシステム管理 (資源管理とネットワークサービス)』の「NFS の障害追跡の方法」を参照してください。

第 31 章

印刷時の問題の解決 (手順)

この章では、印刷サービスの設定または管理の際に発生する可能性のある印刷上の問題を解決する方法について説明します。

この章で説明する手順は次のとおりです。

- 411 ページの「プリンタに出力されない問題を解決する方法」
- 417 ページの「出力が正しくない場合の問題を解決する方法」
- 421 ページの「LP 印刷サービスのハングを解除する方法」
- 422 ページの「アイドル状態になった (ハングした) プリンタの問題を解決する方法」
- 424 ページの「矛盾したプリンタ状態メッセージを解決する方法」

印刷と LP 印刷サービスについては、第 2 章を参照してください。

印刷時の問題解決のヒント

プリンタを設定後に、何も印刷されないことがあります。また、若干は処理されるものの、何か印刷しても正しく出力されない、読みづらいなど、期待どおりの結果が得られないことがあります。このような問題が発生すると、他にも次のような問題が発生することがあります。

- LP コマンドがハングする
- プリンタがアイドル状態になる
- ユーザーが矛盾したメッセージを受け取る

注 - この章の推奨事項の多くはパラレルプリンタに関連しますが、より一般的なシリアルプリンタにも当てはまります。

出力されない (印刷されない) 場合の対処

何も印刷されないときは、次の部分をチェックします。

- プリンタハードウェア
- ネットワーク
- LP 印刷サービス

バナーページは印刷されるのに他には何も印刷されない場合は、不正な出力の特殊ケースです。407 ページの「出力が正しくない場合の対処」を参照してください。

ハードウェアのチェック

ハードウェアは、最初にチェックすべきポイントです。プリンタが電源に接続され、電源がオンになっているかどうかを確認してください。また、ハードウェア付属のマニュアルを参照して、ハードウェアの設定値を調べてください。コンピュータによっては、プリンタポートの特性を変更するハードウェアスイッチが付いているものがあります。

プリンタハードウェアには、プリンタ、コンピュータへの接続ケーブル、ケーブルの先端を接続するポートが含まれます。一般的なアプローチとしては、プリンタからコンピュータへと順番に調べてください。まず、プリンタをチェックします。次に、ケーブルがプリンタに接続される箇所をチェックします。次に、ケーブルをチェックします。最後に、ケーブルがコンピュータに接続されている箇所をチェックします。

ネットワークのチェック

よく問題が発生するのは、印刷クライアントからプリンタサーバーに送られるリモート印刷要求です。プリンタサーバーと印刷クライアント間でネットワークアクセスが使用可能になっているかどうかを確認してください。

ネットワークがネットワーク情報サービスプラス (NIS+) を実行している場合は、システム間のアクセスを有効にする方法について、『Solaris のシステム管理 (ネーミングとディレクトリサービス: FNS, NIS+ 編)』を参照してください。ネットワークがネットワーク情報サービス (NIS) または NIS+ を実行していない場合は、プリンタサーバーと印刷クライアントを設定する前に、プリンタサーバー上の `/etc/hosts` ファイルに各クライアントシステムのインターネットアドレスとシステム名を組み込んでください。また、プリンタサーバーの IP アドレスとシステム名を、各印刷クライアントシステムの `/etc/hosts` ファイルに組み込まなければなりません。

LP 印刷サービスのチェック

正常に印刷するには、プリンタサーバーと印刷クライアント上で LP スケジューラが動作していなければなりません。動作していない場合は、`/usr/lib/lp/lpsched` コマンドを使用して起動する必要があります。スケジューラの起動に問題がある場合は、81 ページの「印刷スケジューラを再起動する方法」を参照してください。

スケジューラが動作している他に、出力する前にプリンタが使用可能になっていて、印刷要求を受け付けられる状態になっていなければなりません。LP 印刷サービスがプリンタへの要求を受け付けなければ、依頼した印刷要求は拒否されます。その場合、一般にユーザーは印刷要求を依頼すると警告メッセージを受け取ります。LP 印刷サービスがプリンタで使用可能になっていないと、印刷要求はプリンタが使用可能になるまでシステム上の待ち行列に残ります。

通常は、次の手順で印刷時の問題を分析してください。

- 手順ごとに印刷要求の経路を追跡します。
- 手順ごとに LP 印刷サービスの状態を調べます。
 - 構成は正しいか
 - プリンタは要求を受け付けるか
 - プリンタは要求を処理できるか
- 要求が転送時にハングしている場合は、`syslog.conf` 内の `lpr.debug` を設定して、転送状況を表示します。
- 要求がローカルでハングしている場合は、`lp sched` ログ (`/var/lp/logs/lpsched`) を調べます。
- 要求がローカルでハングしている場合は、プリンタデバイスエラー (障害) の通知を送らせ、プリンタを再度使用可能にします。

411 ページの「印刷時の問題の解決」に掲載されている手順では、この方法を使用して LP 印刷サービスに関する各種の問題に対処する方法を説明します。

出力が正しくない場合の対処

プリンタと印刷サービスソフトウェアが正しく構成されていない場合は、プリンタで印刷されても、期待どおりに出力されないことがあります。

プリンタタイプとファイル内容形式のチェック

LP 印刷サービスでプリンタを設定するときに間違ったプリンタタイプを使用すると、不適切なプリンタ制御文字がプリンタに送られる可能性があります。その結果は予測できません。何も印刷されない、出力が読みづらい、正しい文字セットやフォントで印刷されないなどの結果となります。

間違ったファイル内容形式を指定した場合、バナーページは印刷できますが、他には何も印刷されません。プリンタに指定されたファイル内容形式は、プリンタがフィルタなしで直接印刷できるファイル形式を示します。ユーザーがプリンタにファイルを送信すると、ファイルはフィルタなしでプリンタに直接送信されます。プリンタがその形式を処理できないときは、問題が発生します。

印刷クライアントの設定時には、ファイル内容形式がプリンタサーバーと印刷クライアントの両方で正しくなければならないので、間違いをおかす機会が多くなります。推奨する方法は、印刷クライアントのファイル内容形式を any に設定することです。こうすると、ファイルはサーバーに直接送信され、フィルタが必要かどうかはサーバー側で決定されます。したがってファイル内容形式は、サーバー側だけで正しく指定すればよいことになります。

印刷クライアント側でファイル内容を指定し、フィルタリングの負荷をサーバーからクライアントに移すことができますが、内容の形式はプリンタサーバー側でサポートしなければなりません。

stty 設定値のチェック

デフォルトの stty (標準端末) 設定値がプリンタから要求される設定値と一致しないと、多数のフォーマット上の問題が生じる可能性があります。この後の節では、設定値の一部が間違っているときに発生する問題について説明します。

ボーレート設定値が正しくない場合

コンピュータのボーレート設定値がプリンタのボーレート設定値と一致しないときは、通常何か出力されますが、希望する出力は得られません。特殊文字や不要なスペースが異常に混じったランダムな出力が表示されます。LP 印刷サービスのデフォルトは 9600 ボーレートです。

注 - プリンタがパラレルポートで接続されている場合、ボーレートの設定値は無関係です。

パリティ設定値が正しくない場合

プリンタによっては、パリティビットを使用して、印刷用に受け取ったデータに伝送中に誤りがなかったことを確認するものがあります。コンピュータとプリンタのパリティビットの設定値は一致しなければなりません。一致しない場合、文字によってはまったく印刷されないか、他の文字で置き換えられることもあります。その出力は文字間隔が正しく、ほとんどの文字が正しい位置にあるので、一見正しいように見えます。LP 印刷サービスの場合、デフォルトではパリティビットは設定されません。

タブ設定値が正しくない場合

ファイルにタブが含まれていても、プリンタがタブを予期していなければ、印刷出力にはファイルの内容が完全に印刷されますが、テキストは右マージンに対して正確に配置されないことがあります。また、プリンタのタブ設定が間違っていると、テキストに左マージンがない、テキストがつながってしまう、テキストがページの一部分に集中する、間違っダブルスペースになってしまうなどの問題が発生します。デフォルトでは、タブは 8 スペースごとに設定されます。

Return 設定値が正しくない場合

出力がシングルスペースのはずなのにダブルスペースになる場合は、プリンタのタブ設定値が間違っているか、プリンタが Return の後に 1 行追加されています。LP 印刷サービスは、改行の前に 1 つ Return を追加するので、その組み合わせによって 2 行の改行が発生します。

ジグザグに印刷される場合は、改行の前に Return を送る stty オプションの onlcr が設定されていません。stty=onlcr オプションはデフォルトで設定されますが、他の印刷問題を解決しようとしたときに、それを消去した可能性があります。

LP コマンドがハングした場合の対処

lp コマンド (lpssystem、lpadmin、lpstat など) を入力しても何も発生しない (エラーメッセージ、状態情報、またはプロンプトが表示されない) 場合は、LP スケジューラに問題が発生した可能性があります。このような問題は、通常は LP スケジューラを停止して再起動すれば解決できます。操作手順については、80 ページの「印刷スケジューラを停止する方法」を参照してください。

プリンタがアイドル状態になった (ハングした) 場合の対処

プリンタが印刷要求を待ち行列に入れているのに、アイドル状態になっていることがあります。プリンタがアイドル状態になっている場合は、次の原因が考えられます。

- 現在の印刷要求にフィルタがかけられている
- プリンタに障害がある
- ネットワーク上の問題が原因で、印刷処理が中断されている

印刷フィルタのチェック

低速印刷フィルタは、プリンタを拘束しないようにバックグラウンドで実行されます。フィルタリングが必要な印刷要求は、フィルタリングが終わるまで印刷されません。

プリンタ障害のチェック

LP 印刷サービスが障害を検出すると、印刷はすぐにではありませんが自動的に再開されます。LP 印刷サービスは約 5 分間待機し、要求が正常に印刷されるまで試行し続けます。プリンタを使用可能にすると、すぐに再試行できます。

ネットワーク上の問題のチェック

ネットワーク経由でファイルを印刷するときには、次の問題が発生することがあります。

- プリンタサーバーに送られた要求が、クライアントシステム (ローカル) の待ち行列で停止する
- プリンタサーバーに送られた要求が、プリンタサーバー (リモート) の待ち行列で停止する

ローカル待ち行列で停止する印刷要求

プリンタサーバーに依頼された印刷要求は、次の原因でクライアントシステムの待ち行列で停止することがあります。

- プリンタサーバーがダウンしている
- プリンタがプリンタサーバー側で使用不可にされている
- 印刷クライアントとプリンタサーバー間のネットワークがダウンしている
- ベースになる互換バージョンのネットワークソフトウェアが適切に設定されていない

問題の原因を突き止めるときには、新しい要求を待ち行列に追加しないでください。詳細は、98 ページの「プリンタへの印刷要求を受け付けるまたは拒否する方法」を参照してください。

リモート待ち行列で停止する印刷要求

印刷要求がプリンタサーバーの待ち行列で停止する場合は、プリンタが使用不可になっている可能性があります。プリンタが要求を受け付けても処理しないとき、その要求は印刷するために待ち行列に入れられます。プリンタを使用可能にすると、それ以外に問題がなければ、待ち行列内の印刷要求は印刷されます。

矛盾した状態メッセージの対処

ユーザーが印刷要求を入力すると、クライアントシステムからは受け付けられたことが通知され、プリンタサーバーからは印刷要求が拒否されたことを示すメールを受け取ることがあります。これらの矛盾したメッセージは、次の原因で発生することがあります。

- 印刷クライアントは要求を受け付けることができても、プリンタサーバーは要求を拒否している場合
- 印刷クライアント側のプリンタの定義が、プリンタサーバー側のプリンタの定義と一致しない場合。特に、フィルタ、文字セット、印字ホイール、フォームなど、印刷ジョブコンポーネントの定義が、クライアントとサーバーシステムの間で一致していない場合

ローカルユーザーがプリンタサーバー上でプリンタにアクセスできるように、これらのジョブコンポーネントの定義が印刷クライアントとプリンタサーバーの両方で登録されているかどうかを確認してください。

印刷時の問題の解決

この節では、次の手順について説明します。

- 出力されない問題を解決する方法
- 出力が正しくない問題を解決する方法
- LP コマンドのハングを解除する方法
- アイドル状態になった (ハングした) プリンタの問題を解決する方法
- 矛盾した状態メッセージを解決する方法

▼ プリンタに出力されない問題を解決する方法

この作業には、次の問題解決の手順が含まれています。印刷要求をプリンタに出したのに何も印刷されない場合は、これらの手順を試してください。

- ハードウェアをチェックする (411 ページの「プリンタハードウェアをチェックする方法」を参照)
- ネットワークをチェックする (412 ページの「プリンタのネットワーク接続をチェックする方法」を参照)
- LP 印刷サービスの基本機能をチェックする (413 ページの「LP 印刷サービスの基本機能をチェックする方法」を参照)
- Solaris 印刷クライアントから Solaris プリンタサーバーへの印刷をチェックする (416 ページの「Solaris 印刷クライアントから Solaris プリンタサーバーへの印刷をチェックする方法」を参照)

印刷クライアント/プリンタサーバーの節に進む前に、上記のうち最初の 3 つの手順をリストの順に試してください。ただし、バナーページは印刷されるが他に何も印刷されない場合は、417 ページの「出力が正しくない場合の問題を解決する方法」の説明に進んでください。

▼ プリンタハードウェアをチェックする方法

1. プリンタがコンセントに接続され、電源がオンになっているか確認します。
2. ケーブルがプリンタのポートと、システムまたはサーバーのポートに接続されているか確認します。
3. そのケーブルが正しいケーブルであり、欠陥がないことを確認します。

詳細は、ハードウェア付属のマニュアルを参照してください。プリンタがシリアルポートに接続されている場合は、そのケーブルでハードウェアフロー制御がサポートされることを確認してください。ヌルモデムアダプタでは、この機能がサポートされます。次の表は、ヌルモデムケーブル用のピン構成を示しています。

表 31-1 ヌルモデムケーブル用のピン構成

	ホスト	プリンタ
Mini-Din-8	25-Pin D-sub	25-Pin D-sub
-	1(FG)	1(FG)
3(TD)	2(TD)	3(RD)
5(RD)	3(RD)	2(TD)
6(RTS)	4(RTS)	5(CTS)
2(CTS)	5(CTS)	4(RTS)
4(SG)	7(SG)	7(SG)
7(DCD)	6(DSR)、8(DCD)	20(DTR)
1(DTR)	20(DTR)	6(DSR)、8(DCD)

4. ポート用のハードウェアスイッチが正しく設定されていることを確認します。
正しい設定については、プリンタのマニュアルを参照してください。
5. プリンタが動作するか確認します。
プリンタにセルフテスト機能が付いている場合は、その機能を使用します。プリンタのセルフテストの詳細は、プリンタのマニュアルを参照してください。
6. コンピュータとプリンタのボーレートの設定値が正しいか確認します。
コンピュータとプリンタのボーレートの設定値が一致しなければ、何も印刷されないことがあり、さらに正しく出力されない場合もあります。詳細は、417 ページの「出力が正しくない場合の問題を解決する方法」を参照してください。

▼ プリンタのネットワーク接続をチェックする方法

1. ping コマンドを使用すると、プリンタサーバーと印刷クライアント間のネットワークが正しく設定されているか確認できます。

```
print_client# ping print_server
print_server is alive
print_server# ping print_client
print_client not available
```

システムが動作していることを示すメッセージが表示されれば、そのシステムにアクセスできることがわかるので、そのネットワークは正常です。また、このメッセージは、入力したホスト (システム) 名が、ネームサーバーまたはローカルの /etc/hosts ファイルによって IP アドレスに変換されたことを示します。変換されていない場合は、IP アドレスを入力する必要があります。

「not available」というメッセージが表示された場合は、次の 3 点を確認してください。まず、NIS または NIS+ はサイトでどのように設定されているか。次に、プ

プリンタサーバーと印刷クライアントが相互に通信できるように付加的な作業が必要か。最後に、サイトがNISまたはNIS+を実行していない場合、各印刷クライアントの/etc/hostsファイルにプリンタサーバーのIPアドレスを入力し、プリンタサーバーの/etc/hostsファイルにすべての印刷クライアントのIPアドレスを入力したか確認します。

2. (SunOS 5.0 – 5.1 プリンタサーバーのみ) listen ポートモニターが正しく構成されているか確認します。
3. (SunOS 5.0 – 5.1 プリンタサーバーのみ) ネットワーク待機サービスがプリンタサーバー上のポートモニターに登録されているか確認します。

▼ LP 印刷サービスの基本機能をチェックする方法

この手順では、基本 LP 印刷サービス機能をチェックする例として、プリンタ luna を使用しています。

1. プリンタサーバー上で、LP 印刷サービスが動作していることを確認します。
 - a. このコマンドは、LP スケジューラが動作しているか表示します。

```
# lpstat -r
scheduler is running
```
 - b. スケジューラが動作していない場合は、スーパーユーザーまたは lp になり、スケジューラを起動します。

```
# /usr/lib/lp/lpsched
```

スケジューラを起動できない場合は、421 ページの「LP 印刷サービスのハングを解除する方法」を参照してください。
2. プリンタサーバー上と印刷クライアント上で、プリンタが要求を受け付けていることを確認します。
 - a. プリンタが要求を受け付けていることを確認します。

```
# lpstat -a
mars accepting requests since Jul 12 14:23 2001
luna not accepting requests since Jul 12 14:23 2001
unknown reason
```

このコマンドは、LP システムがシステム用に構成された各プリンタの要求を受け付けているか確認します。
 - b. プリンタが要求を受け付けていない場合は、スーパーユーザーまたは lp になり、プリンタが印刷要求を受け付けるようにします。

```
# accept luna
```

これで、指定したプリンタは要求を受け付けます。

3. プリンタサーバー上と印刷クライアント上で、プリンタが依頼された印刷要求の印刷で使用可能になっているか確認します。

- a. プリンタが使用可能になっていることを確認します。

```
# lpstat -p luna
printer luna disabled since Jul 12 14:25 2001.
available.
unknown reason
```

このコマンドは、プリンタの状態に関する情報を表示します。プリンタ名を省略すると、システム用に設定されたすべてのプリンタに関する情報を表示できます。次の例は、使用不可になっているプリンタを示しています。

- b. プリンタが使用不可になっている場合は、スーパーユーザーまたは lp になり、プリンタを使用可能にします。

```
# enable luna
printer "luna" now enabled.
```

指定したプリンタが、印刷要求の処理に使用可能になります。

4. プリンタサーバー上で、プリンタが正しいシリアルポートに接続されていることを確認します。

- a. プリンタが正しいシリアルポートに接続されていることを確認します。

```
# enable luna
printer "luna" now enabled.
```

「device for printer-name」というメッセージは、ポートアドレスを示します。LP 印刷サービスの接続先のポートにケーブルが接続されているか確認します。ポートが正しければ、手順 5 に進みます。

- b. スーパーユーザーまたは lp になります。

- c. ポートを表すデバイスファイルのファイル所有権を変更します。

```
# chown lp device-filename
```

このコマンドは、特殊なユーザー lp をデバイスファイルの所有者として割り当てます。このコマンドで、*device-filename* はデバイスファイル名です。

- d. プリンタポートのデバイスファイルのアクセス権を変更します。

```
# chmod 600 device-filename
```

このコマンドにより、スーパーユーザーまたは lp だけがプリンタポートデバイスファイルにアクセスできます。

5. プリンタサーバー上と印刷クライアント上で、プリンタが正しく構成されていることを確認します。

- a. プリンタが適切に設定されていることを確認します。

```
# lpstat -p luna -l
printer luna is idle. enabled since Jul 12 14:24 2001. available
```

```
Content types: postscript
Printer types: PS
```

上の例は、正しく設定された PostScript プリンタと、そのプリンタを印刷要求の処理に利用できることを示しています。プリンタタイプとファイル内容形式が正しい場合は、手順 6 に進みます。

- b. プリンタタイプまたはファイル内容形式が違っている場合は、印刷クライアント上で、プリンタタイプを unknown に設定し、内容形式を any に設定してください。

```
# lpadmin -p printer-name -T printer-type -I file-content-type
```

6. プリンタサーバー上で、プリンタがプリンタ障害のために待機していないことを確認します。

- a. プリンタ障害のためにプリンタが待機していないことを確認します。

```
# lpadmin -p printer-name -F continue
```

このコマンドは LP 印刷サービスに対して、障害のために待機していない場合は続行するように指示します。

- b. プリンタを再び使用可能にすることによって、すぐに再試行させます。

```
# enable printer-name
```

- c. (省略可能) プリンタ障害をすぐに通知するように、LP 印刷サービスに指示します。

```
# lpadmin -p printer-name -A 'write root'
```

このコマンドは LP 印刷サービスに対して、プリンタが障害を起こした場合に、root に書き込むというデフォルトポリシーを設定し、root がログインした端末にプリンタ障害メッセージを送るように指示します。これにより、問題を修正するときに障害通知をすぐに受け取れます。

7. プリンタがログイン端末として間違った設定になっていないか確認します。

注 – ログイン端末としてプリンタを設定する作業では誤りをおかしやすいので、当てはまらないと思われる場合にも、必ず設定値を確認してください。

- a. ps -ef コマンドの出力で、プリンタポートのエントリを探します。

```
# ps -ef
root 169 167 0 Apr 04 ? 0:08 /usr/lib/saf/listen tcp
root 939 1 0 19:30:47 ? 0:02 /usr/lib/lpsched
root 859 858 0 19:18:54 term/a 0:01 /bin/sh -c \ /etc/lp/
interfaces/luna
luna-294 rocket!smith "passwd\n##
#
```

このコマンドの出力で、プリンタポートのエントリを探します。上の例で、ポート /dev/term/a はログイン端末として間違っ設定されています。この行の最後に "passwd\n## 情報が付いているのでわかります。ポートが正しく設定されている場合は、この手順の最後を飛ばしてください。

- b. 印刷要求を取り消します。

```
# cancel request-id
```

このコマンドで、*request-id* は取り消したい印刷要求の要求 ID 番号です。

- c. プリンタポートをログインデバイス以外のものとして設定します。

```
# lpadmin -p printer-name -h
```

- d. `ps -ef` コマンドからの出力をチェックして、プリンタポートがログインデバイスではなくなったことを確認します。

基本的な LP 印刷サービス機能に印刷時の問題の原因が見つからない場合は、次の中から該当するクライアント/サーバーの手順に進んでください。

▼ Solaris 印刷クライアントから Solaris プリンタサーバーへの印刷をチェックする方法

1. まだチェックしていなければ、プリンタサーバー上で LP 印刷サービスの基本機能をチェックします。

基本機能をチェックする手順については、413 ページの「LP 印刷サービスの基本機能をチェックする方法」を参照してください。印刷クライアントから要求が出されたときに何も印刷されない原因を調べる前に、プリンタがローカルで動作していることを確認してください。

2. まだチェックしていなければ、印刷クライアント上で LP 印刷サービスの基本機能をチェックします。

印刷クライアントから印刷を要求する前に、そのプリンタを有効にして、その要求を受け付けるようにする必要があります。

注 - 次の手順のほとんどは、root または lp としてログインして実行しなければなりません。

3. プリンタサーバーがアクセス可能であることを確認します。

- a. 印刷クライアント上で、`ping print-server` と入力して Return キーを押します。このコマンドにより、プリンタサーバーに応答を求める要求が送られます。

```
print_client# ping print_server
```

「*print_server* not available」というメッセージが表示される場合は、ネットワークに問題があります。

4. プリンタサーバーが正常に動作しているか確認します。

```
# lpstat -t luna
scheduler is running
system default destination: luna
device for luna: /dev/term/a
luna accepting requests since Jul 12 14:26 2001
printer luna now printing luna-314. enabled since Jul 12 14:26 2001.
available.
luna-129          root          488    Jul 12 14:32
#
```

上記の例は、プリンタサーバーが動作していることを示します。

5. プリンタサーバーが正常に動作していない場合は、手順 1 に戻ります。

▼ 出力が正しくない場合の問題を解決する方法

1. スーパーユーザーまたは lp としてログインします。
2. プリンタタイプが正しいことを確認します。

プリンタタイプが正しくないと、正しく出力されないことがあります。たとえば、プリンタタイプ PS を指定してもページが意図した順序の逆に印刷される場合は、プリンタタイプ PSR を試してください(この 2 つのタイプ名は大文字で指定しなければなりません)。また、プリンタタイプが正しくないと、テキストの欠落、読みづらいテキスト、または間違ったフォントのテキストが出力されることがあります。プリンタタイプを判別するには、terminfo データベース内のエントリを調べます。terminfo データベースの構造については、48 ページの「プリンタタイプ」を参照してください。

- a. プリンタサーバー上で、プリンタの特性を表示します。

```
$ lpstat -p luna -l
printer luna is idle. enabled since Mon Jul 12 15:02:32 ...
  Form mounted:
  Content types: postscript
  Printer types: PS
  Description:
  Connection: direct
  Interface: /usr/lib/lp/model/standard
  After fault: continue
  Users allowed:
    (all)
  Forms allowed:
    (none)
  Banner not required
  Character sets:

  Default pitch:
  Default page size: 80 wide 66 long
  Default port settings:

$
```

- b. プリンタのマニュアルを参照して、プリンタのモデルを調べます。
- c. プリンタタイプが正しくない場合は、**Solaris** プリンタマネージャの「プリンタのプロパティを変更 (**Modify Printer Properties**)」ウィンドウを使用して変更するか、次の `lpadmin` コマンドを使用します。

```
# lpstat -p printer-name -T printer-type
```

印刷クライアント上では、プリンタタイプを `unknown` にしてください。プリンタサーバー上では、プリンタタイプは使用するプリンタのモデルをサポートするように定義された `terminfo` エントリと一致しなければなりません。使用するプリンタのタイプに関する `terminfo` エントリがない場合は、138 ページの「サポートされていないプリンタの `terminfo` エントリを追加する方法」を参照してください。

- 3. バナーページは印刷されるが文書の本文が印刷されない場合は、ファイル内容形式を確認します。

プリンタに指定したファイル内容形式は、プリンタがフィルタなしで直接印刷できるファイル形式を示します。ファイル内容形式が正しくなければ、必要なときにフィルタリングがバイパスされることがあります。

- a. 前の手順の `lpstat` コマンドで表示されたファイル内容形式に関する情報をメモします。

印刷クライアント上では、1つ以上の明示的な内容形式を指定する理由がない限り、ファイル内容形式を `any` にしてください。クライアント上で内容を指定すると、プリンタサーバー上ではなく印刷クライアント上でフィルタリングが実行されます。また、クライアント上の内容形式は、プリンタサーバー上で指定した内容形式と一致しなければならず、プリンタサーバー上の内容形式はプリンタの機能を反映していなければなりません。

- b. プリンタのマニュアルを参照し、プリンタで直接印刷できるファイルのタイプを判別します。

これらのファイル形式を参照するために使用する名前は、プリンタメーカーが使用している名前と一致しなくてもかまいません。ただし、使用する名前は LP 印刷サービスに認識されるフィルタで使用する名前と一致しなければなりません。

- c. ファイル内容形式が正しくない場合は、**Solaris** プリンタマネージャの「プリンタのプロパティを変更 (**Modify Printer Properties**)」ウィンドウを使用して変更するか、次の `lpadmin` コマンドを使用します。

```
# lpadmin -p printer-name -I file-content-type(s)
```

必要に応じて、このコマンドを印刷クライアント上、プリンタサーバー上、またはその両方で実行します。印刷クライアント上で `-I any` を試し、プリンタサーバー上で `-I ""` を試してください。 `-I ""` は、NULL のファイル内容形式リストを指定します。これは、プリンタはそのプリンタタイプと正確に一致するファイルしか直接印刷できないので、すべてのファイルをフィルタにかけることを意味します。

ファイルが印刷されないときは、まずこの組み合わせを選択してみるとよいでしょう。それで成功したら、プリンタサーバー上で明示的な内容形式を指定し、不要なフィルタリングを減らすことができます。ローカルの PostScript プリンタで

は、プリンタでサポートされている場合は、`postscript` または `postscript, simple` を使用してください。PS と PSR はファイル内容形式ではなく、プリンタタイプなので注意してください。

`-I` を省略すると、ファイル内容のリストはデフォルトの `simple` になります。`-I` オプションを使用し、`simple` 以外にもファイル内容形式を指定したい場合は、リストに `simple` を含めなければなりません。

複数のファイル内容形式を指定するときは、名前をコンマで区切ります。また、名前をスペースで区切り、リストを引用符で囲むこともできます。ファイル内容形式として `any` を指定すると、フィルタリングは行われないので、プリンタで直接印刷できるファイルタイプのみを送信する必要があります。

4. フォントのダウンロードに必要なフィルタリングを、印刷要求がバイパスしていないかどうかをチェックします。

ユーザーがコマンド `lp -T PS` を使用して印刷要求を PostScript プリンタに依頼すると、フィルタリングは実行されません。フィルタリングを強制するコマンド `lp -T postscript` を使用して要求を依頼しようとする、文書に必要な非常駐フォントがダウンロードされることがあります。

5. プリンタポートの `stty` 設定値が正しいことを確認します。
 - a. プリンタのマニュアルを参照して、プリンタポートに合った `stty` 設定値を判別します。

注 – プリンタがパラレルポートまたは USB ポートで接続されている場合、ポートの設定値は無関係です。

- b. 現在の設定値を調べるには、`stty` コマンドを使用します。

```
# stty -a < /dev/term/a
speed 9600 baud;
rows = 0; columns = 0; ypixels = 0; xpixels = 0;
eucw 1:0:0:0, scrw 1:0:0:0
intr = ^c; quit = ^|; erase = ^?; kill = ^u;
eof = ^d; eol = <undef>; eol2 = <undef>; swtch = <undef>;
start = ^q; stop = ^s; susp = ^z; dsusp = ^y;
rprnt = ^r; flush = ^o; werase = ^w; lnext = ^v;
parenb -parodd cs7 -cstopb -hupcl cread -clocal -loblk -parext
-ignbrk brkint -ignpar -parmrk -inpck istrib -inlcr -igncr icrnl -iuclc
ixon -ixany -ixoff imaxbel
isig icanon -xcase echo echoe echok -echonl -noflsh
-tostop echoctl -echoprt echoke -defecho -flusho -pendin iexten
opost -olcuc onlcr -ocrnl -onocr -onlret -ofill -ofdel tab3
#
```

このコマンドでは、プリンタポートの現在の `stty` 設定値が表示されます。

LP 印刷サービスの標準プリンタインタフェースプログラムで使用されるデフォルトの `stty` オプションを表 31-2 に示します。

表 31-2 標準インタフェースプログラムで使用されるデフォルト stty 設定値

オプション	説明
-9600	ボーレートを 9600 に設定
-cs8	8 ビットバイトを設定
-cstopb	1 バイト当たり 1 ストップビットを送信
-parity	パリティを生成しない
-ixon	XON/XOFF (START/STOP または DC1/DC3 ともいう) を使用可能にする
-opost	以下にリストされた設定値をすべて使用して「処理後出力」を実行する
-olcuc	小文字を大文字に割り当てない
-onlcr	改行をキャリッジリターン/改行に変更する
-ocrnl	キャリッジリターンを改行に変更しない
-onocr	カラム 0 でもキャリッジリターンを出力する
-nl0	改行後の遅延なし
-cr0	キャリッジリターン後の遅延なし
-tab0	タブ後の遅延なし
-bs0	バックスペース後の遅延なし
-vt0	垂直タブ後の遅延なし
-ff0	用紙送り後の遅延なし

c. stty 設定値を変更します。

```
# lpadmin -p printer-name -o "stty= options"
```

表 31-3 を使用して、印刷出力に影響する様々な問題を解決する stty オプションを選択します。

表 31-3 印刷出力の問題を解決する stty オプション

stty 値	結果	間違っただ設定から起こり得る問題
110, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400	ボーレートを指定した値に設定する (ボーレートを 1 つだけ入力する)	ランダム文字と特殊文字が印刷され、間隔がバラバラになることがある

表 31-3 印刷出力の問題を解決する stty オプション (続き)

stty 値	結果	間違っただ設定から起こり得る問題
oddp	奇数パリティを設定する	文字が欠落または間違っただ文字がランダムに表示される
evenp	偶数パリティを設定する	
-parity	パリティを設定しない	
-tabs	タブを設定しない	テキストが右マージンにくっついてしまう
tabs	8 スペースごとにタブを設定する	テキストに左マージンがなく、つながってしまうか、くっついてしまう
-onlcr	行頭でキャリッジリターンを設定しない	間違っただダブルスペース
onlcr	行頭でキャリッジリターンを設定する	ジグザグに印刷される

オプションをスペースで区切り、オプションリストを単一引用符で囲むと、複数のオプションの設定を変更できます。たとえば、奇数パリティを使用可能にし、7 ビットの文字サイズを設定する必要があるプリンタを仮定します。そのためには、次の例のようなコマンドを入力します。

```
# lpadmin -p neptune -o "stty='parenb parodd cs7'"
```

stty オプション parenb でパリティチェック/生成を使用可能にし、parodd で奇数パリティの生成を設定し、cs7 で文字サイズを7ビットに生成します。

6. 文書が正しく印刷されることを確認します。

```
# lp -d printer-name filename
```

▼ LP 印刷サービスのハングを解除する方法

1. スーパーユーザーまたは lp としてログインします。
2. LP 印刷サービスを停止します。

```
# /etc/init.d/lp stop
```

3. LP 印刷サービスを再起動します。

```
# /etc/init.d/lp start
```

LP 印刷サービスが再起動されます。スケジューラが再起動されない場合は、81 ページの「印刷スケジューラを再起動する方法」を参照してください。

▼ アイドル状態になった (ハングした) プリンタの問題を解決する方法

この作業には、プリンタがアイドル状態であってはならないのにアイドル状態になるときに使用する多数の手順が含まれています。通常は各手順を順番に試しますが、順番どおりでなくてもかまいません。

▼ プリンタの準備ができていないかチェックする方法

1. プリンタ状態情報を表示します。

```
# lpstat -p printer-name
```

表示される情報は、プリンタがアイドル状態かアクティブ状態か、使用可能か使用不可か、または印刷要求を利用できるか受け付けていないかを示します。すべて正常と思われる場合は、この節の他の手順に進んでください。lpstat コマンドを実行できない場合は、421 ページの「LP 印刷サービスのハングを解除する方法」を参照してください。

2. プリンタが利用できない (要求を受け付けていない) 場合は、プリンタが要求を受け付けるようにします

```
# accept printer-name
```

プリンタは、その印刷待ち行列に要求を受け付け始めます。

3. プリンタが使用不可になっている場合は、再び使用可能にします。

```
# enable printer-name
```

このコマンドでは、待ち行列にある要求を処理するように、プリンタを再び使用可能にします。

▼ 印刷のフィルタリングをチェックする方法

lpstat -o コマンドを使用して、印刷のフィルタリングをチェックします。

```
$ lpstat -o luna
luna-10      fred      1261    Mar 12 17:34 being filtered
luna-11      iggy      1261    Mar 12 17:36 on terra
luna-12      jack      1261    Mar 12 17:39 on terra
$
```

待機している最初の要求がフィルタリングされているかどうかを調べます。上の例のような出力になる場合は、ファイルがフィルタリングされています。プリンタはハングせず、要求の処理に少し時間がかかっているだけです。

▼ プリンタ障害の後に印刷を再開する方法

1. プリンタ障害に関するメッセージがある場合は、その障害を解決してください。
プリンタ障害の警告がどのように指定されているかに応じて、メッセージを電子メールで root に送らせるか、root がログインした端末に書き出すことができます。
2. プリンタを再び使用可能にします。

```
# enable printer-name
```

プリンタ障害によって要求がブロックされた場合は、このコマンドで強制的に再試行します。このコマンドが動作しない場合は、この節の他の手順を続行します。

▼ ローカル待ち行列で停止している印刷要求をリモートプリンタに送信する方法

1. 印刷クライアント上で、プリンタサーバーへの印刷要求を、それ以上待ち行列に入れないようにします。

```
# reject printer-name
```

2. 印刷クライアント上で、プリンタサーバーに **ping** 要求 (存在をチェックする要求) を送信します。

```
print_client# ping print_server  
print_server is alive
```

「*print_server not available*」というメッセージが表示される場合は、ネットワークに問題があります。

3. 問題を解決したら、新しい印刷要求を待ち行列に入れられるようにします。

```
# accept printer-name
```

4. 必要であれば、再びプリンタを使用可能にします。

```
# enable printer-name
```

▼ プリンタサーバーの待ち行列で停止する印刷クライアントからの印刷要求を使用可能にする方法

1. プリンタサーバー上で、印刷クライアントからプリンタサーバーへの印刷要求を、それ以上待ち行列に入れないようにします。

```
# reject printer-name
```

2. `lpsched` ログファイルを表示します。

```
# more /var/lp/logs/lpsched
```

表示される情報を参考にして、印刷クライアントからプリンタサーバーへの印刷要求が印刷されない原因を正確に把握できます。

3. 問題を解決したら、新しい印刷要求を待ち行列に入れられるようにします。

```
# accept printer-name
```

4. 必要であれば、プリンタサーバー上で再びプリンタを使用可能にします。

```
# enable printer-name
```

▼ 矛盾したプリンタ状態メッセージを解決する方法

1. プリンタサーバー上でプリンタが使用可能になっており、要求を受け付けているかどうかを確認します。

```
# lpstat -p printer-name
```

印刷クライアントが要求を受け付けているのに、プリンタサーバーが要求を拒否しているときは、矛盾した状態メッセージが表示されます。

2. プリンタサーバー上で、印刷クライアント上のプリンタの定義が、プリンタサーバー上のプリンタの定義と一致するかどうかを確認します。

```
# lpstat -p -l printer-name
```

印刷フィルタ、文字セット、印字ホイール、フォームなど、印刷ジョブコンポーネントの定義を調べて、印刷クライアントとサーバー上で一致し、ローカルユーザーがプリンタサーバーシステムのプリンタにアクセスできることを確認します。

第 32 章

UFS ファイルシステムの不整合解決 (手順)

この章では、`fsck` エラーメッセージとエラーメッセージの解決に回答できる内容について説明します。

この章で説明する情報は次のとおりです。

- 427 ページの「`fsck` の一般エラーメッセージ」
- 428 ページの「初期化フェーズでの `fsck` メッセージ」
- 431 ページの「フェーズ 1: ブロックとサイズの検査のメッセージ」
- 434 ページの「フェーズ 1B: 走査し直して DUPS メッセージを表示する」
- 434 ページの「フェーズ 2: パス名の検査のメッセージ」
- 440 ページの「フェーズ 3: 接続性の検査のメッセージ」
- 442 ページの「フェーズ 4: 参照数の検査のメッセージ」
- 445 ページの「フェーズ 5: シリンダグループ検査のメッセージ」
- 446 ページの「クリーンアップ (後処理) フェーズのメッセージ」

`fsck` コマンドと、そのコマンドを使用してファイルシステムの完全性を確認する方法については、『*Solaris* のシステム管理 (基本編)』の「UFS ファイルシステムの整合性チェック (手順)」を参照してください。

`fsck` エラーメッセージ

通常、システムが異常終了し、ファイルシステムの最新の変更がディスクに書き込まれなかった場合に、`fsck` コマンドが非対話形式で実行され、ファイルシステムが修復されます。ファイルシステムの基本的な非整合状態は自動的に修正されますが、より重大なエラーは修復されません。ファイルシステムを修復する間に、`fsck` コマンドはこの種の異常終了から予想される非整合状態を修正します。より重大な状況の場合は、エラーが表示されて終了します。

fsck コマンドを対話形式で実行すると、見つかった各非整合状態を表示して小さなエラーを修正します。ただし、より重大なエラーの場合は、非整合状態を表示し、応答を選択するように促します。-y または -n オプションを指定して fsck コマンドを実行する場合、それぞれのエラー条件に対する fsck コマンドのデフォルトの応答は、yes または no に置き換えられます。

修正処置によっては、若干のデータが失われます。失われるデータの量は、fsck の診断出力から判断できます。

fsck コマンドはファイルシステムの複数の段階による検査プログラムです。各段階ごとに、fsck コマンドの異なるフェーズを呼び出し、メッセージも異なります。初期化後に、fsck コマンドはファイルシステムごとに各パスを逐次実行して、ブロックとサイズ、パス名、接続状態、参照数、空きブロックマップを検査します (再構築することもあります)。また、後処理も実行します。

UFS バージョンの fsck コマンドによって実行されるフェーズ (段階) は次のとおりです。

- 初期化
- フェーズ 1- ブロックとサイズの検査
- フェーズ 2- パス名の検査
- フェーズ 3- 接続状態の検査
- フェーズ 4- 参照数の検査
- フェーズ 5- シリンダグループの検査

この後の各節では、各フェーズで検出できるエラー条件、表示されるメッセージとプロンプト、および応答できる内容について説明します。

複数のフェーズで表示されるメッセージについては、427 ページの「fsck の一般エラーメッセージ」を参照してください。それ以外の場合、メッセージは発生するフェーズのアルファベット順に掲載されています。

fsck エラーメッセージには、次の表に示す省略形が含まれています。

表 32-1 エラーメッセージの省略形

省略形	説明
BLK	ブロック番号
DUP	重複ブロック番号
DIR	ディレクトリ名
CG	シリンダグループ
MTIME	ファイルの最終変更時刻
UNREF	非参照

また、多くのメッセージには、i ノード番号などの変数フィールドが含まれています。このマニュアルでは、i ノード番号を *inode-number* のようにイタリック体で記載してあります。たとえば、次の画面メッセージは、

```
INCORRECT BLOCK COUNT I=2529
```

次の例のように記載されています。

```
INCORRECT BLOCK COUNT I=inode-number
```

fsck の一般エラーメッセージ

この節のエラーメッセージは、初期化後のどのフェーズでも表示されることがあります。処理を続けるかどうかのオプションは表示されますが、通常は、致命的だと見なすのが最善の処置です。これらのエラーメッセージは重大なシステム障害を反映しており、ただちに処理する必要があります。この種のメッセージが表示された場合は、n(o) を入力してプログラムを終了してください。問題の原因を判断できない場合は、ご購入先に問い合わせてください。

```
CANNOT SEEK: BLK block-number (CONTINUE)
```

エラーの発生原因

ファイルシステム内で、指定されたブロック番号 *block-number* へ移動させるという要求に失敗した。このメッセージは重大な問題、おそらくハードウェア障害を示す。

ファイルシステムのチェックを続けると、fsck は移動を再び行い、移動できなかったセクタ番号のリストを表示する。ブロックが仮想メモリーバッファークャッシュの一部であれば、fsck は致命的な入出力エラーメッセージを表示して終了する。

対処方法

ディスクにハードウェア障害が発生していると、この問題は継続する。もう一度 fsck を実行してファイルシステムをチェックする。

このチェックでも解決しない場合、ご購入先に問い合わせる。

```
CANNOT READ: BLK block-number (CONTINUE)
```

エラーの発生原因

ファイルシステム内で指定されたブロック番号 *block-number* を読み込むという要求に失敗した。このメッセージは重大な問題、おそらくハードウェア障害を示す。

ファイルシステムのチェックを続けたい場合、fsck は読み取りを再試行して、読み込めなかったセクター番号のリストを表示する。ブロックが仮想メモリーバッファークャッシュの一部であれば、fsck は致命的な入出力エラーメッセージを表示して終了する。fsck が読み取りに失敗したブロックのいずれかに書き込もうとすると、次のメッセージが表示される。

```
WRITING ZERO'ED BLOCK sector-numbers TO DISK
```

対処方法

ディスクにハードウェア障害が発生していると、この問題は継続する。もう一度 `fsck` を実行してファイルシステムをチェックする。このチェックでも解決しない場合、購入先に問い合わせる。

```
CANNOT WRITE: BLK block-number (CONTINUE)
```

エラーの発生原因

ファイルシステム内で、指定されたブロック番号 *block-number* への書き込みに失敗した。

ファイルシステムのチェックを続けると、`fsck` は書き込みを再度実行し、書き込めなかったセクタ番号のリストを表示する。ブロックが仮想メモリーバッファークャッシュの一部であれば、`fsck` は致命的な入出力エラーメッセージを表示して終了する。

対処方法

ディスクが書き込み保護されている可能性がある。ドライブ上で書き込み保護ロックをチェックする。ディスクにハードウェア障害がある場合、問題は解決しない。もう一度 `fsck` を実行してファイルシステムをチェックする。書き込み保護が原因でない場合、あるいはファイルシステムを再チェックしても問題が解決しない場合は、購入先に問い合わせる。

初期化フェーズでの `fsck` メッセージ

初期化フェーズでは、コマンド行構文がチェックされます。ファイルシステムのチェックを実行する前に、`fsck` はテーブルを設定してファイルを開きます。

この節のメッセージは、コマンド行オプション、メモリー要求、ファイルのオープン、ファイルの状態、ファイルシステムのサイズチェック、およびスクラッチファイルの作成によるエラー条件に関するものです。ファイルシステムを修復する間に、どんな初期化エラーが発生した場合も、`fsck` は終了します。

```
Can't roll the log for device-name.
```

```
DISCARDING THE LOG MAY DISCARD PENDING TRANSACTIONS.
```

```
DISCARD THE LOG AND CONTINUE?
```

エラーの発生原因

UFS ファイルシステムのエラーチェックを行う前に、記録されている UFS ファイルシステムのトランザクションログをフラッシュできなかった。

対処方法

`yes` と応答する場合には、ファイルシステム操作がログに記録されていても、ファイルシステムに適用されなければ、そのファイルシステム操作は失われる。この場合、`fsck` は、通常と同様の検査を実行してフェーズ 5 で次の質問が表示される。

```
FREE BLK COUNT(S) WRONG IN SUPERBLK (SALVAGE)
```

ここで yes と応答すると、ログに使用したブロックを回収する。次回にログを有効にしてファイルシステムをマウントすると、ログが再作成される。

no と応答すると、ログを保持したまま終了するが、ファイルシステムはマウントできない。

```
bad inode number inode-number to ginode
```

エラーの発生原因

inode-number が存在しないため、内部エラーが発生した。fsck は終了する。

対処方法

ご購入先に問い合わせる。

```
cannot alloc size-of-block map bytes for blockmap
```

```
cannot alloc size-of-free map bytes for freemap
```

```
cannot alloc size-of-state map bytes for statemap
```

```
cannot alloc size-of-lncntp bytes for lncntp
```

エラーの発生原因

内部テーブル用のメモリー要求に失敗した。fsck は終了する。このメッセージは、即座に処理しなければならない重大なシステム障害を示す。他のプロセスが大量のシステム資源を使用していると、このエラー条件が発生することがある。

対処方法

他のプロセスを終了すると問題を解決できることがある。解決できない場合は、ご購入先に問い合わせる。

```
Can't open checklist file: filename
```

エラーの発生原因

ファイルシステムの検査リストファイル *filename* (通常は /etc/vfstab) を開いて読み込めない。fsck は終了する。

対処方法

ファイルの有無と、そのアクセスモードで読み取りが可能かどうかを検査する。

```
Can't open filename
```

エラーの発生原因

fsck はファイルシステム *filename* を開けなかった。対話形式で実行している場合、fsck はこのファイルシステムを無視し、次に指定されたファイルシステムの検査を続ける。

対処方法

そのファイルシステムの raw デバイスファイルに読み取り、または書き込みができるかどうかをチェックする。

```
Can't stat root
```

エラーの発生原因

fsck はルートディレクトリに関する統計情報要求に失敗した。fsck は終了する。

対処方法

このメッセージは、重大なシステム障害を示す。ご購入先に問い合わせる。

```
Can't stat filename
Can't make sense out of name filename
```

エラーの発生原因

fsck はファイルシステム *filename* に関する統計情報要求に失敗した。対話形式で実行している場合、fsck はこのファイルシステムを無視し、次に指定されたファイルシステムの検査を続ける。

対処方法

ファイルシステムの有無とそのアクセスモードをチェックする。

```
filename: (NO WRITE)
```

エラーの発生原因

-n オプションが指定されているか、fsck はファイルシステム *filename* を書き込み用に開けなかった。fsck を非書き込みモードで実行中であれば、診断メッセージはすべて表示されるが、fsck は何も修正しない。

対処方法

-n を指定しなかった場合は、指定したファイルのタイプを検査する。通常ファイル名の可能性がある。

```
IMPOSSIBLE MINFREE=percent IN SUPERBLOCK (SET TO DEFAULT)
```

エラーの発生原因

スーパーブロックの最小容量が 99 パーセントを超えているか、0 パーセント未満である。

対処方法

minfree パラメータをデフォルトの 10 パーセントに設定するには、デフォルトプロンプトから *y* と入力する。エラー条件を無視するには、デフォルトプロンプトから *n* と入力する。

```
filename: BAD SUPER BLOCK: message
USE AN ALTERNATE SUPER-BLOCK TO SUPPLY NEEDED INFORMATION;
e.g., fsck[-f ufs] -o b=# [special ...]
where # is the alternate superblock. See fsck_ufs(1M)
```

エラーの発生原因

スーパーブロックが破損している。

対処方法

次のいずれかのメッセージが表示される。

```
CPG OUT OF RANGE
FRAGS PER BLOCK OR FRAGSIZE WRONG
INODES PER GROUP OUT OF RANGE
INOPB NONSENSICAL RELATIVE TO BSIZE
MAGIC NUMBER WRONG
NCG OUT OF RANGE
NCYL IS INCONSISTENT WITH NCG*CPG
NUMBER OF DATA BLOCKS OUT OF RANGE
NUMBER OF DIRECTORIES OUT OF RANGE
ROTATIONAL POSITION TABLE SIZE OUT OF RANGE
SIZE OF CYLINDER GROUP SUMMARY AREA WRONG
SIZE TOO LARGE
```

BAD VALUES IN SUPERBLOCK

代替スーパーブロックを使用して `fsck` を再実行してみる。手始めにブロック 32 を指定するとよい。スライスに対して `newfs -N` コマンドを実行すると、スーパーブロックの代替コピーの位置を調べることができる。-N を指定しないと、`newfs` は既存のファイルシステムを上書きするので注意する。

UNDEFINED OPTIMIZATION IN SUPERBLOCK (SET TO DEFAULT)

エラーの発生原因

スーパーブロックの最適化パラメータが `OPT_TIME` でも `OPT_SPACE` でもない。

対処方法

ファイルシステム上で処理の実行時間を最小限度まで短縮するには、`SET TO DEFAULT` プロンプトから `y` を入力する。このエラー条件を無視するには、`n` と入力する。

フェーズ 1: ブロックとサイズの検査のメッセージ

このフェーズでは、`i` ノードリストを検査します。次の処理中に検出されたエラー条件が表示されます。

- `i` ノードのタイプを検査する
- ゼロリンク数テーブルを設定する
- 不良ブロックまたは重複ブロックの有無を `i` ノードブロック番号で検査する
- `i` ノードのサイズを検査する
- `i` ノードの形式を検査する

ファイルシステムの修復 (`preen`) 中は、`INCORRECT BLOCK COUNT`、`PARTIALLY TRUNCATED INODE`、`PARTIALLY ALLOCATED INODE`、および `UNKNOWN FILE TYPE` を除き、このフェーズ中にどのエラーが発生した場合も、`fsck` が終了します。

フェーズ 1 では、次のメッセージ (アルファベット順) が発生する可能性があります。

```
block-number BAD I=inode-number
```

エラーの発生原因

`i` ノード `inode-number` に、ファイルシステム内の最初のデータブロックより小さい番号または最後のデータブロックより大きい番号が付いたブロック番号 `block-number` が入っている。`i` ノード `inode-number` 内にファイルシステムの範囲外のブロック番号が多すぎると、このエラー条件のためにフェーズ 1 で「`EXCESSIVE BAD BLKS`」エラーメッセージが生成されることがある。フェーズ 2 と 4 では、このエラー条件が原因で「`BAD/DUP`」エラーメッセージが生成される。

対処方法

なし

```
BAD MODE: MAKE IT A FILE?
```

エラーの発生原因

指定された *i* ノードの状態がすべて、ファイルシステムの損傷を示す 1 に設定されている。このメッセージは、`fsck -y` が実行された後で繰り返し表示される場合以外は、物理的なディスクの損傷を示すものではない。

対処方法

y と入力して *i* ノードを妥当な値に初期化し直す。

```
BAD STATE state-number TO BLKERR
```

エラーの発生原因

内部エラーによって `fsck` の状態マップが破壊されたため、不可能な値 *state-number* を示す。`fsck` は即座に終了する。

対処方法

ご購入先に問い合わせる。

```
block-number DUP I=inode-number
```

エラーの発生原因

i ノード *inode-number* には、同じ *i* ノードまたは別の *i* ノードがすでに取得したブロック番号 *block-number* が入っている。このエラー条件が発生した場合に、*i* ノード *inode-number* 内にこの種のブロック番号が多すぎると、フェーズ 1 では「EXCESSIVE DUP BLKS」エラーメッセージが生成されることがある。このエラー条件によってフェーズ 1B が呼び出され、フェーズ 2 と 4 で「BAD/DUP」エラーメッセージが生成される。

対処方法

なし

```
DUP TABLE OVERFLOW (CONTINUE)
```

エラーの発生原因

`fsck` の内部テーブルには、重複するブロック番号が入る余地がない。`-o p` (`preen`、修復) オプションが指定されていると、プログラムは終了する。

対処方法

プログラムを続行するには、`CONTINUE` プロンプトから *y* と入力する。このエラーが発生すると、ファイルシステムを完全には検査できない。別の重複ブロックが見つかり、このエラー条件が再発する。使用可能な仮想メモリの容量を (プロセスを終了し、スワップ空間を拡張して) 大きくし、もう一度 `fsck` を実行してファイルシステムを検査し直す。プログラムを終了するには *n* と入力する。

```
EXCESSIVE BAD BLOCKS I=inode-number (CONTINUE)
```

エラーの発生原因

i ノード *inode-number* に関連付けられたファイルシステム内の最初のデータブロックより小さい番号か、最後のブロックより大きい番号を持つブロックが多すぎる (通常は 10 以上)。`-o p` (`preen`、修復) オプションを指定すると、プログラムは終了する

対処方法

プログラムを続行するには、`CONTINUE` プロンプトから *y* と入力する。このエラーが発生すると、ファイルシステムを完全には検査できない。もう一度 `fsck` を実行

してファイルシステムを検査し直す必要がある。プログラムを終了するには `n` と入力する。

EXCESSIVE DUP BLKS I=*inode-number* (CONTINUE)

エラーの発生原因

同じ `i` ノード、別の `i` ノード、または空きリストが取得するブロック数が多すぎる (通常は 10 以上)。 `-o p` (`preen`、修復) オプションを指定すると、プログラムは終了する。

対処方法

プログラムを続行するには、CONTINUE プロンプトから `y` と入力する。このエラーが発生すると、ファイルシステムを完全には検査できない。もう一度 `fsck` を実行してファイルシステムを検査し直す必要がある。プログラムを終了するには `n` と入力する。

INCORRECT BLOCK COUNT I=*inode-number* (*number-of-BAD-DUP-or-missing-blocks* should be *number-of-blocks-in-filesystem*) (CORRECT)

エラーの発生原因

`i` ノード *inode-number* のブロック数は *number-of-BAD-DUP-or-missing-blocks* であるが、*number-of-blocks-in-filesystem* でなければならない。修復 (`preen`) の場合、`fsck` は数を訂正する。

対処方法

`i` ノード *inode-number* のブロック数を *number-of-blocks-in-filesystem* に置き換えるには、CORRECT プロンプトから `y` と入力する。プログラムを終了するには `n` と入力する。

LINK COUNT TABLE OVERFLOW (CONTINUE)

エラーの発生原因

`fsck` の内部テーブルには、リンク数が 0 の割り当て済み `i` ノードが入る余地がない。 `-o p` (`preen`、修復) オプションを指定すると、プログラムは終了するので、`fsck` を手作業で実行する必要がある。

対処方法

プログラムを続行するには、CONTINUE プロンプトから `y` と入力する。リンク数が 0 の別の割り当て済みブロックが見つかったら、このエラー条件が再発する。このエラーが発生すると、ファイルシステムを完全には検査できない。もう一度 `fsck` を実行してファイルシステムを検査し直す必要がある。プロセスをいくつか終了するか、スワップ領域を拡張して、使用可能な仮想メモリーを増やしてから、`fsck` を実行し直す。プログラムを終了するには `n` と入力する。

PARTIALLY ALLOCATED INODE I=*inode-number* (CLEAR)

エラーの発生原因

`i` ノード *inode-number* は割り当て済みでも未割り当てでもない。 `-o p` (`preen`、修復) オプションを指定すると、この `i` ノードは消去される。

対処方法

`i` ノード *inode-number* の内容を消去して割り当てを解除するには、`y` と入力する。これにより、この `i` ノードを指すディレクトリごとに、フェーズ 2 でエラー条件 UNALLOCATED が生成されることがある。このエラー条件を無視するには、`n` と入

力する。応答しなくてよいのは、この問題を他の手段で解決しようとする場合だけである。

PARTIALLY TRUNCATED INODE I=*inode-number* (SALVAGE)

エラーの発生原因

`fsck` で、割り当てられたブロック数よりも短い *i* ノード *inode-number* が見つかった。この条件が発生するのは、ファイルの切り捨て中にシステムがクラッシュした場合だけである。ファイルシステムを修復しているとき、`fsck` は指定されたサイズへの切り捨てを完了する。

対処方法

i ノード内で指定したサイズへの切り捨てを完了するには、`SALVAGE` プロンプトから `y` と入力する。このエラー条件を無視するには、`n` と入力する。

UNKNOWN FILE TYPE I=*inode-number* (CLEAR)

エラーの発生原因

i ノード *inode-number* のモードのワードは、この *i* ノードがパイプ、特殊文字 *i* ノード、特殊ブロック *i* ノード、通常 *i* ノード、シンボリックリンク、FIFO ファイル、またはディレクトリ *i* ノードでないことを示す。-`op` (`preen`、修復) オプションを指定すると、この *i* ノードは消去される。

対処方法

i ノード *inode-number* の内容を消去して割り当てを解除するには、`CLEAR` プロンプトから `y` と入力する。これにより、この *i* ノードを指すディレクトリエントリごとに、フェーズ 2 でエラー条件 `UNALLOCATED` が生成される。このエラー条件を無視するには、`n` と入力する。

フェーズ 1B: 走査し直して DUPS メッセージを表示する

ファイルシステム内で重複ブロックが見つかったら、次のメッセージが表示されます。

```
block-number DUP I=inode-number
```

エラーの発生原因

i ノード *inode-number* には、すでに同じ *i* ノードまたは別の *i* ノードによって取得されたブロック番号 *block-number* が入っている。このエラー条件によって、フェーズ 2 で `BAD/DUP` エラーメッセージが生成される。重複ブロックを持つ *i* ノードは、このエラー条件とフェーズ 1 の `DUP` エラー条件を検査すれば判断できる。

対処方法

重複ブロックが見つかったら、ファイルシステムが再び走査され、以前にそのブロックを取得した *i* ノードが検索される。

フェーズ 2: パス名の検査のメッセージ

このフェーズでは、フェーズ 1 と 1B で見つかった不良 *i* ノードを指すディレクトリエントリが削除される。次の原因によるエラー条件が表示されます。

- 不正なルート i ノードモードと状態
- 範囲外のディレクトリ i ノードポインタ
- 不良 i ノードを指すディレクトリエントリ
- ディレクトリ完全性検査

ファイルシステムを修復している場合は (-o p (preen、修復) オプション)、このフェーズでどのエラーが発生した場合も、fsck が終了します。ただし、ブロックサイズの倍数でないディレクトリ、重複ブロックと不良ブロック、範囲外の i ノード、過剰なハードリンクに関連するエラーは除きます。

フェーズ 2 では、次のメッセージ (アルファベット順) が表示される可能性があります。

```
BAD INODE state-number TO DESCEND
```

エラーの発生原因

fsck の内部エラーによって、ファイルシステムのディレクトリ構造を継承するルーチンに、無効な状態 *state-number* が渡された。fsck は終了する。

対処方法

このエラーメッセージが表示される場合は、ご購入先に問い合わせる。

```
BAD INODE NUMBER FOR '.' I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

エラーの発生原因

「.」の i ノード番号が *inode-number* に等しくないディレクトリ *inode-number* が見つかった。

対処方法

「.」の i ノード番号を *inode-number* に等しくなるように変更するには、FIX プロンプトから *y* と入力する。「.」の i ノード番号を変更しない場合は、*n* と入力する。

```
BAD INODE NUMBER FOR '..' I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

エラーの発生原因

「..」の i ノード番号が *inode-number* の親に等しくないディレクトリ *inode-number* が見つかった。

対処方法

「..」の i ノード番号を *inode-number* の親に等しくなるように変更するには、FIX プロンプトから *y* と入力する (ルート i ノード内の「..」は、それ自体を指すので注意する)。「..」の i ノード番号を変更しない場合は、*n* と入力する。

```
BAD RETURN STATE state-number FROM DESCEND
```

エラーの発生原因

fsck の内部エラーによって、ファイルシステムのディレクトリ構造を継承するルーチンから、不可能な状態 *state-number* が返された。fsck は終了する。

対処方法

このメッセージが表示される場合は、ご購入先に問い合わせる。

```
BAD STATE state-number FOR ROOT INODE
```

エラーの発生原因

内部エラーによって、ルート *i* ノードに不可能な状態 *state-number* が割り当てられた。fsck は終了する。

対処方法

このエラーメッセージが表示される場合は、ご購入先に問い合わせる。

```
BAD STATE state-number FOR INODE=inode-number
```

エラーの発生原因

内部エラーによって、*i* ノード *inode-number* に不可能な状態 *state-number* が割り当てられた。fsck は終了する。

対処方法

このエラーメッセージが表示される場合は、ご購入先に問い合わせる。

```
DIRECTORY TOO SHORT I=inode-number OWNER=UID MODE=file-mode  
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

エラーの発生原因

サイズ *file-size* が最小ディレクトリサイズより小さいディレクトリ *filename* が見つかった。所有者 *UID*、モード *file-mode*、サイズ *file-size*、変更時刻 *modification-time*、およびディレクトリ名 *filename* が表示される。

対処方法

ディレクトリのサイズを最小ディレクトリサイズまで大きくするには、FIX プロンプトから *y* と入力する。このディレクトリを無視するには *n* と入力する。

```
DIRECTORY filename: LENGTH file-size NOT MULTIPLE OF block-number (ADJUST)
```

エラーの発生原因

サイズ *file-size* がディレクトリブロックのサイズ *block-number* の倍数でないディレクトリ *filename* が見つかった。

対処方法

長さを適切なブロックサイズに切り上げるには、*y* と入力する。ファイルシステムを修復しているとき (-o p (preen、修復)、オプション) は、fsck は警告のみを表示してディレクトリを調整する。この条件を無視するには *n* と入力する。

```
DIRECTORY CORRUPTED I=inode-number OWNER=UID MODE=file-mode  
SIZE=file-size MTIME=modification-time DIR=filename (SALVAGE)
```

エラーの発生原因

内部状態の整合性がないディレクトリが見つかった。

対処方法

次のディレクトリ境界 (通常は 512 バイトの境界) までのすべてのエントリを放棄するには、SALVAGE プロンプトから *y* と入力する。この処置によって、最高で 42 個のエントリを放棄できる。この処置は、他の回復作業に失敗した場合にのみ実行する。問題のディレクトリを変更せずに、次のディレクトリ境界までスキップして読み取りを再開するには、*n* と入力する。

```
DUP/BAD I=inode-number OWNER=O MODE=M SIZE=file-size
MTIME=modification-time TYPE=filename (REMOVE)
```

エラーの発生原因

フェーズ 1 またはフェーズ 1B で、ディレクトリまたはファイルエントリ *filename*、*i* ノード *inode-number* に関連付けられた重複ブロックまたは不良ブロックが見つかった。所有者 *UID*、モード *file-mode*、サイズ *file-size*、変更時刻 *modification-time*、ディレクトリまたはファイル名 *filename* が表示される。-p (preen、修復) オプションを指定すると、重複または不良ブロックが削除される。

対処方法

ディレクトリまたはファイルのエントリ *filename* を削除するには、REMOVE プロンプトから *y* と入力する。このエラー条件を無視するには、*n* と入力する。

```
DUPS/BAD IN ROOT INODE (REALLOCATE)
```

エラーの発生原因

フェーズ 1 またはフェーズ 1B で、ファイルシステムのルート *i* ノード (通常は *i* ノード番号 2) に、重複ブロックまたは不良ブロックが見つかった。

対処方法

ルート *i* ノードの既存の内容を消去して再割り当てを行うには、REALLOCATE プロンプトから *y* と入力する。一般にルート内で検出されるファイルとディレクトリがフェーズ 3 で復元され、lost+found ディレクトリに格納される。ルートの割り当てに失敗すると、fsck は「CANNOT ALLOCATE ROOT INODE」というメッセージを表示して終了する。*n* と入力すると、CONTINUE プロンプトが表示される。*y* と入力して CONTINUE プロンプトに回答するとルート *i* ノード内の DUPS/BAD エラー条件を無視して、ファイルシステムの検査を続行する。ルート *i* ノードが不正であれば、他の多数のエラーメッセージが生成されることがある。*n* の場合は、プログラムを終了する。

```
EXTRA '.' ENTRY I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

エラーの発生原因

「.」のエントリが複数個入っているディレクトリ *inode-number* が見つかった。

対処方法

「.」の余分なエントリを削除するには、FIX プロンプトから *y* と入力する。問題のディレクトリを変更しない場合は、*n* と入力する。

```
EXTRA '..' ENTRY I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

エラーの発生原因

「..」(親ディレクトリ)のエントリが複数個入っているディレクトリ *inode-number* が見つかった。

対処方法

「..」(親ディレクトリ)の余分なエントリを削除するには、FIX プロンプトから *y* と入力する。問題のディレクトリを変更しない場合は、*n* と入力する。

```
hard-link-number IS AN EXTRANEIOUS HARD LINK TO A DIRECTORY filename (REMOVE)
```

エラーの発生原因

fsckによって、ディレクトリ *filename* へのハードリンク *hard-link-number* にエラーが見つかった。修復 (preen) しているとき (-o p オプション)、fsck はエラーのあるハードリンクを無視する。

対処方法

エラーのあるエントリ *hard-link-number* を削除するには、プロンプトから *y* と入力する。このエラー条件を無視するには、*n* と入力する。

```
inode-number OUT OF RANGE I=inode-number NAME=filename (REMOVE)
```

エラーの発生原因

ディレクトリエントリ *filename* には、i ノードリストの終わりより大きい i ノード番号 *inode-number* が付いている。-p (preen、修復) オプションを指定すると、i ノードが自動的に削除される。

対処方法

ディレクトリエントリ *filename* を削除するには、REMOVE プロンプトから *y* と入力する。このエラー条件を無視するには、*n* と入力する。

```
MISSING '.' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size  
MTIME=modification-time DIR=filename (FIX)
```

エラーの発生原因

最初のエントリ (「.」のエントリ) に未割り当てのディレクトリ *inode-number* が見つかった。

対処方法

i ノード番号が *inode-number* に等しい「.」のエントリを構築するには、FIX プロンプトから *y* と入力する。問題のディレクトリを変更しない場合は、*n* と入力する。

```
MISSING '.' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size  
MTIME=modification-time DIR=filename CANNOT FIX, FIRST ENTRY IN  
DIRECTORY CONTAINS filename
```

エラーの発生原因

最初のエントリが *filename* となっているディレクトリ *inode-number* が見つかった。fsck はこの問題を解決できない。

対処方法

このエラーメッセージが表示される場合は、ご購入先に問い合わせる。

```
MISSING '.' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size  
MTIME=modification-time DIR=filename CANNOT FIX, INSUFFICIENT  
SPACE TO ADD '.'
```

エラーの発生原因

最初のエントリが「.」でないディレクトリ *inode-number* が見つかった。fsck は問題を解決できない。

対処方法

このエラーメッセージが表示される場合は、ご購入先に問い合わせる。

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size  
MTIME=modification-time DIR=filename (FIX)
```

エラーの発生原因

第2のエントリが割り当てられていないディレクトリ *inode-number* が見つかった。

対処方法

i ノード番号が *inode-number* の親に等しい「*..*」のエントリを構築するには、FIX プロンプトから *y* と入力する (ルート *i* ノード内の「*..*」は、それ自体を指すので注意する)。問題のディレクトリを変更しない場合は、*n* と入力する。

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, SECOND ENTRY IN
DIRECTORY CONTAINS filename
```

エラーの発生原因

第2のエントリが *filename* となっているディレクトリ *inode-number* が見つかった。
fsck はこの問題を解決できない。

対処方法

このエラーメッセージが表示される場合は、ご購入先に問い合わせる。

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, INSUFFICIENT SPACE
TO ADD '..'
```

エラーの発生原因

第2のエントリが「*..*」(親ディレクトリ) でないディレクトリ *inode-number* が見つかった。*fsck* はこの問題を解決できない。

対処方法

このエラーメッセージが表示される場合は、ご購入先に問い合わせる。

```
NAME TOO LONG filename
```

エラーの発生原因

長すぎるパス名が見つかった。通常、これはファイルシステムの名前空間内のループを示す。特権を持つユーザーがディレクトリへの循環リンクを作成すると、このエラーが発生することがある。

対処方法

循環リンクを削除する。

```
ROOT INODE UNALLOCATED (ALLOCATE)
```

エラーの発生原因

ルート *i* ノード (通常は *i* ノード番号 2) に割り当てモードビットがない。

対処方法

i ノード 2 をルート *i* ノードとして割り当てるには、ALLOCATE プロンプトから *y* と入力する。一般にルート内で検出されるファイルとディレクトリがフェーズ 3 で復元され、lost+found ディレクトリに格納される。ルートの割り当てに失敗すると、*fsck* は「CANNOT ALLOCATE ROOT INODE」というメッセージを表示して終了する。プログラムを終了するには *n* と入力する。

```
ROOT INODE NOT DIRECTORY (REALLOCATE)
```

エラーの発生原因

ファイルシステムのルート *i* ノード (通常は *i* ノード番号 2) はディレクトリ *i* ノードではない。

対処方法

ルート *i* ノードの既存の内容を消去して再割り当てを行うには、REALLOCATE プロンプトから *y* と入力する。一般にルート内で検出されるファイルとディレクトリがフェーズ 3 で復元され、lost+found ディレクトリに格納される。ルートの割り当てに失敗すると、fsck は「CANNOT ALLOCATE ROOT INODE」というメッセージを表示して終了する。fsck に FIX プロンプトを表示させるには、*n* と入力する。

```
UNALLOCATED I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time type=filename (REMOVE)
```

エラーの発生原因

ディレクトリまたはファイルのエントリ *filename* は、未割り当ての *i* ノード *inode-number* を指している。所有者 *UID*、モード *file-mode*、サイズ *file-size*、変更時刻 *modification-time*、およびファイル名 *filename* が表示される。

対処方法

ディレクトリエントリ *filename* を削除するには、REMOVE プロンプトから *y* と入力する。このエラー条件を無視するには、*n* と入力する。

```
ZERO LENGTH DIRECTORY I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (REMOVE)
```

エラーの発生原因

ディレクトリエントリ *filename* のサイズ *file-size* が 0 になっている。所有者 *UID*、モード *file-mode*、サイズ *file-size*、変更時刻 *modification-time*、およびディレクトリ名 *filename* が表示される。

対処方法

ディレクトリエントリ *filename* を削除するには、REMOVE プロンプトから *y* と入力する。これにより、フェーズ 4 で「BAD/DUP」エラーメッセージが表示される。このエラー条件を無視するには、*n* と入力する。

フェーズ 3: 接続性の検査のメッセージ

このフェーズでは、フェーズ 2 で検査したディレクトリが検査され、次の原因によるエラー条件が表示されます。

- 参照されないディレクトリ
- lost+found ディレクトリが見つからないまたは一杯

フェーズ 3 では、次のメッセージがアルファベット順に表示される可能性があります。

```
BAD INODE state-number TO DESCEND
```

エラーの発生原因

内部エラーによって、ファイルシステムのディレクトリ構造を継承するルーチンに、不可能な状態 *state-number* が渡された。fsck は終了する。

対処方法

このエラーが発生する場合は、ご購入先に問い合わせる。

```
DIR I=inode-number1 CONNECTED. PARENT WAS I=inode-number2
```

エラーの発生原因

これは、ディレクトリ *i* ノード *inode-number1* が *lost+found* ディレクトリに正常に接続されていることを示す。ディレクトリ *i* ノード *inode-number1* の親 *i* ノード *inode-number2* は、*lost+found* ディレクトリの *i* ノード番号に置き換えられる。

対処方法

なし

```
DIRECTORY filename LENGTH file-size NOT MULTIPLE OF block-number (ADJUST)
```

エラーの発生原因

サイズ *file-size* がディレクトリのブロックサイズ *B* の倍数でないディレクトリ *filename* が見つかった (この条件は、フェーズ 2 で調整しなければ、フェーズ 3 で再発することがある)。

対処方法

長さを適切なブロックサイズまで切り上げるには、ADJUST プロンプトから *y* と入力する。修復しているときは、*fsck* は警告を表示してディレクトリを調整する。このエラー条件を無視するには、*n* と入力する。

```
lost+found IS NOT A DIRECTORY (REALLOCATE)
```

エラーの発生原因

lost+found のエントリがディレクトリではない。

対処方法

ディレクトリ *i* ノードを割り当てて、それを参照する *lost+found* ディレクトリを変更するには、REALLOCATE プロンプトから *y* と入力する。以前に *lost+found* ディレクトリによって参照されていた *i* ノードは消去されず、非参照の *i* ノードとして再び取得されるか、このフェーズの後半でそのリンク数が調整される。*lost+found* ディレクトリを作成できない場合は、「SORRY. CANNOT CREATE *lost+found* DIRECTORY」というメッセージが表示され、消失 *i* ノードへのリンク試行が中止される。このエラーにより、フェーズ 4 で UNREF エラーメッセージが生成される。フェーズ 4 で UNREF エラーメッセージを生成する消失 *i* ノードへのリンク試行を中止するには、*n* と入力する。

```
NO lost+found DIRECTORY (CREATE)
```

エラーの発生原因

ファイルシステムのルートディレクトリ内に *lost+found* ディレクトリがない。修復するときに、*fsck* は *lost+found* ディレクトリを作成しようとする。

対処方法

ファイルシステムのルート内で *lost+found* ディレクトリを作成するには、CREATE プロンプトから *y* と入力する。このため、「NO SPACE LEFT IN / (EXPAND)」というメッセージが表示されることがある。*lost+found* ディレクトリを作成できない場合は、*fsck* によって「SORRY. CANNOT CREATE *lost+found* DIRECTORY」というメッセージが表示され、消失 *i* ノードへのリンク試行が中止される。これにより、フェーズ 4 の後半で UNREF エラーメッセージ

が生成される。消失した *i* ノードへのリンク試行を中止するには、*n* と入力する。

NO SPACE LEFT IN /lost+found (EXPAND)

エラーの発生原因

使用可能な領域がないため、ファイルシステムのルートディレクトリ内で、lost+found ディレクトリに別のエントリを追加できない。修復するときに、fsck は lost+found ディレクトリを拡張する。

対処方法

lost+found ディレクトリを拡張して新しいエントリを追加する余地をつくるには、EXPAND プロンプトから *y* と入力する。拡張試行に失敗すると、fsck によって「SORRY. NO SPACE IN lost+found DIRECTORY」というメッセージが表示され、lost+found ディレクトリへのファイルリンク要求が中止される。このエラーにより、フェーズ 4 の後半で UNREF エラーメッセージが生成される。lost+found ディレクトリ内で不要なエントリを削除する。修復中にこのエラーが起きると、fsck は終了する。消失 *i* ノードへのリンク試行を中止するには、*n* と入力する。

UNREF DIR I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
MTIME=*modification-time* (RECONNECT)

エラーの発生原因

ファイルシステムの走査中に、ディレクトリ *i* ノード *inode-number* がディレクトリ エントリに接続されなかった。ディレクトリ *i* ノード *inode-number* の所有者 *UID*、モード *file-mode*、サイズ *file-size*、および変更時刻 *modification-time* が表示される。修復しているときは、ディレクトリサイズが 0 でなければ、fsck は空でないディレクトリ *i* ノードを接続し直す。それ以外の場合、fsck はディレクトリ *i* ノードを消去する。

対処方法

ディレクトリ *i* ノード *inode-number* を lost+found ディレクトリに接続し直すには、RECONNECT プロンプトから *y* と入力する。ディレクトリが再び正常に接続されると、「CONNECTED」というメッセージが表示される。それ以外の場合は、lost+found エラーメッセージのいずれかが表示される。このエラー条件を無視するには、*n* と入力する。このエラーにより、フェーズ 4 で UNREF エラー条件が発生する。

フェーズ 4: 参照数の検査のメッセージ

このフェーズでは、フェーズ 2 と 3 で取得したリンク数情報が検査されます。次の原因によるエラー条件が表示されます。

- 非参照ファイル
- lost+found ディレクトリが見つからないまたは一杯
- ファイル、ディレクトリ、シンボリックリンク、または特殊ファイルの不正なリンク数
- 非参照ファイル、シンボリックリンク、ディレクトリ
- ファイルとディレクトリ内の不良ブロックまたは重複ブロック
- 不正な合計空き *i* ノード数

このフェーズのすべてのエラー (lost+found ディレクトリ内の容量不足を除く) は、ファイルシステムを修復するときに解決できます。

フェーズ 4 では、次のメッセージ (アルファベット順) が表示される可能性があります。

```
BAD/DUP type I=inode-number OWNER=UID MODE=file-mode SIZE=file-size  
MTIME=modification-time (CLEAR)
```

エラーの発生原因

フェーズ 1 またはフェーズ 1B で、ファイルまたはディレクトリ *i* ノード *inode-number* に関連付けられた重複ブロックまたは不良ブロックが見つかった。 *i* ノード *inode-number* の所有者 *UID*、モード *file-mode*、サイズ *file-size*、および変更時刻 *modification-time* が表示される。

対処方法

i ノード *inode-number* の内容を消去して割り当てを解除するには、CLEAR プロンプトから *y* と入力する。このエラー条件を無視するには、*n* と入力する。

(CLEAR)

エラーの発生原因

直前の UNREF エラーメッセージで記述された *i* ノードを再び接続できない。ファイルシステムを修復していると、ファイルを接続し直すには容量が足りないため *fsck* が終了するので、このメッセージは表示されない。

対処方法

i ノードの内容を消去して割り当てを解除するには、CLEAR プロンプトから *y* と入力する。直前のエラー条件を無視するには、*n* と入力する。

```
LINK COUNT type I=inode-number OWNER=UID MODE=file-mode  
SIZE=file-size  
MTIME=modification-time COUNT link-count SHOULD BE  
corrected-link-count (ADJUST)
```

エラーの発生原因

ディレクトリまたはファイル *i* ノード *inode-number* のリンク数は *link-count* になっているが、*corrected-link-count* でなければならない。 *i* ノード *inode-number* の所有者 *UID*、モード *file-mode*、サイズ *file-size*、および変更時刻 *modification-time* が表示される。 *-o p* (*preen*、修復) オプションを指定すると、参照数が増えていない限り、リンク数が調整される。この条件は、ハードウェア障害がなければ発生しない。参照数が修復中に増えると、*fsck* は「LINK COUNT INCREASING」というメッセージを表示して終了する。

対処方法

ディレクトリまたはファイル *i* ノード *inode-number* のリンク数を *corrected-link-count* に置き換えるには、ADJUST プロンプトから *y* と入力する。このエラー条件を無視するには、*n* と入力する。

```
lost+found IS NOT A DIRECTORY (REALLOCATE)
```

エラーの発生原因

lost+found のエントリがディレクトリではない。

対処方法

ディレクトリ *i* ノードを割り当てて、それを参照する `lost+found` ディレクトリを変更するには、`REALLOCATE` プロンプトから `y` と入力する。`lost+found` による以前の *i* ノード参照は消去されない。非参照 *i* ノードとして再び取得されるか、そのリンク数がこのフェーズの後半で調整される。`lost+found` ディレクトリを作成できない場合は、「`SORRY. CANNOT CREATE lost+found DIRECTORY`」というメッセージが表示され、消失 *i* ノードへのリンク試行が中止される。このエラーにより、フェーズ 4 の後半で `UNREF` エラーメッセージが生成される。消失 *i* ノードへのリンク試行を中止するには、`n` と入力する。

`NO lost+found DIRECTORY (CREATE)`

エラーの発生原因

ファイルシステムのルートディレクトリ内に `lost+found` ディレクトリがない。修復するときに、`fsck` は `lost+found` ディレクトリを作成しようとする。

対処方法

ファイルシステムのルート内で `lost+found` ディレクトリを作成するには、`CREATE` プロンプトから `y` と入力する。`lost+found` ディレクトリを作成できない場合は、`fsck` によって「`SORRY. CANNOT CREATE lost+found DIRECTORY`」というメッセージが表示され、消失 *i* ノードへのリンク試行が中止される。このエラーにより、フェーズ 4 の後半で `UNREF` エラーメッセージが生成される。消失 *i* ノードへのリンク試行を中止するには、`n` と入力する。

`NO SPACE LEFT IN / lost+found (EXPAND)`

エラーの発生原因

ファイルシステムのルートディレクトリ内で、`lost+found` ディレクトリに別のエントリを追加する容量がない。修復するときに、`fsck` は `lost+found` ディレクトリを拡張する。

対処方法

`lost+found` ディレクトリを拡張して新しいエントリを追加する余地をつくるには、`EXPAND` プロンプトから `y` と入力する。拡張試行に失敗すると、`fsck` によって「`SORRY. NO SPACE IN lost+found DIRECTORY`」というメッセージが表示され、`lost+found` ディレクトリへのファイルリンク要求が中止される。このエラーにより、フェーズ 4 の後半で `UNREF` エラーメッセージが生成される。`lost+found` ディレクトリ内で不要なエントリを削除する。修復 (`-o p` オプション) が有効なときは、このエラーによって `fsck` が終了する。消失 *i* ノードへのリンク試行を中止するには、`n` と入力する。

`UNREF FILE I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time (RECONNECT)`

エラーの発生原因

ファイルシステムを走査したときに、ファイル *i* ノード `inode-number` がディレクトリエントリに接続されなかった。*i* ノード `inode-number` の所有者 `UID`、モード `file-mode`、サイズ `file-size`、および変更時刻 `modification-time` が表示される。`fsck` が修復しているときに、ファイルのサイズまたはリンク数が 0 であれば、そのファイルは消去される。それ以外の場合は再び接続される。

対処方法

i ノード *inode-number* を `lost+found` ディレクトリ内のファイルシステムに接続し直すには、`y` と入力する。i ノード *inode-number* を `lost+found` ディレクトリに接続できないと、このエラーによってフェーズ 4 で `lost+found` エラーメッセージが生成されることがある。このエラー条件を無視するには、`n` と入力する。このエラーが発生すると、フェーズ 4 で必ず `CLEAR` エラー条件が呼び出される。

```
UNREF type I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time (CLEAR)
```

エラーの発生原因

ファイルシステムを走査するときに、i ノード *inode-number* (その *type* はディレクトリまたはファイル) がディレクトリエントリに接続されなかった。i ノード *inode-number* の所有者 *UID*、モード *file-mode*、サイズ *file-size*、および変更時刻 *modification-time* が表示される。`fsck` が修復しているときに、ファイルのサイズまたはリンク数が 0 であれば、そのファイルは消去される。それ以外の場合は再び接続される。

対処方法

i ノード *inode-number* の内容を消去して割り当てを解除するには、`CLEAR` プロンプトから `y` と入力する。このエラー条件を無視するには、`n` と入力する。

```
ZERO LENGTH DIRECTORY I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time (CLEAR)
```

エラーの発生原因

ディレクトリエントリ *filename* のサイズ *file-size* が 0 になっている。所有者 *UID*、モード *file-mode*、サイズ *file-size*、変更時刻 *modification-time*、およびディレクトリ名 *filename* が表示される。

対処方法

i ノード *inode-number* の内容を消去して割り当てを解除するには、`y` と入力する。このエラー条件を無視するには、`n` と入力する。

フェーズ 5: シリンダグループ検査のメッセージ

このフェーズでは、空きブロックと使用済み i ノードのマップが検査されます。次の原因によるエラー条件が表示されます。

- 使用済み i ノードマップから欠落している割り当て済み i ノード
- 空きブロックマップから欠落している空きブロック
- 使用済み i ノードマップ内の空き i ノード
- 不正な合計空きブロック数
- 不正な合計使用済み i ノード数

フェーズ 5 では、次のメッセージがアルファベット順に表示される可能性があります。

```
BLK(S) MISSING IN BIT MAPS (SALVAGE)
```

エラーの発生原因

シリンダグループのブロックマップから空きブロックがいくつか欠落している。修復中に、`fsck` はマップを作成し直す。

対処方法

空きブロックマップを作成し直すには、`SALVAGE` プロンプトから `y` と入力する。このエラー条件を無視するには、`n` と入力する。

`CG character-for-command-option: BAD MAGIC NUMBER`

エラーの発生原因

シリンダグループ `character-for-command-option` のマジック番号が間違っている。通常、このエラーはシリンダグループマップが破壊されていることを示す。対話形式で実行している場合は、シリンダグループに再度の作成が必要であることを示すマークが付けられる。ファイルシステムを修復している場合は、`fsck` が終了する。

対処方法

このエラーが発生する場合は、ご購入先に問い合わせる。

`FREE BLK COUNT(S) WRONG IN SUPERBLK (SALVAGE)`

エラーの発生原因

空きブロック数の実際のが、ファイルシステムのスーパーブロック内の空きブロック数と一致しない。`-o p` (`preen`、修復) オプションを指定した場合は、スーパーブロック内の空きブロック数が自動的に修正される。

対処方法

スーパーブロックの空きブロック情報を作成し直すには、`SALVAGE` プロンプトから `y` と入力する。このエラー条件を無視するには、`n` と入力する。

`SUMMARY INFORMATION BAD (SALVAGE)`

エラーの発生原因

集計情報が間違っている。修復していると、`fsck` は集計情報を計算し直す。

対処方法

集計情報を作成し直すには、`SALVAGE` プロンプトから `y` と入力する。このエラー条件を無視するには、`n` と入力する。

クリーンアップ (後処理) フェーズのメッセージ

ファイルシステムのチェックが終わると、クリーンアップ処理がいくつか実行されます。クリーンアップフェーズでは、次の状態メッセージが表示されます。

`number-of files, number-of-files used, number-of-files free (number-of frags, number-of blocks, percent fragmentation)`

上記のメッセージは、検査されたファイルシステムに、フラグメントサイズの *number-of* 個のブロックを使用中の *number-of* 個のファイルが入っていることと、ファイルシステム内でフラグメントサイズのブロックが *number-of* 個空いていることを示します。括弧内の数は、空いている数を *number-of* 個の空きフラグメント、*number-of* 個の完全サイズの空きブロック、および *percent* のフラグメントに分割したものです。

```
***** FILE SYSTEM WAS MODIFIED *****
```

上記のメッセージは、ファイルシステムが `fsck` によって変更されたことを示します。このファイルシステムがマウントされているか、現在のルート (/) ファイルシステムの場合はリブートします。ファイルシステムがマウントされている場合は、マウント解除して再び `fsck` を実行する必要があります。そうしないと、`fsck` によって実行された処理がテーブルのインコアコピー (カーネル内のコピー) によって取り消されます。

```
filename FILE SYSTEM STATE SET TO OKAY
```

上記のメッセージは、ファイルシステム *filename* に安定を示す印が付けられたことを示します。-m オプションを指定して `fsck` を実行すると、この情報を使用して、ファイルシステムの検査が必要かどうか判断されます。

```
filename FILE SYSTEM STATE NOT SET TO OKAY
```

上記のメッセージは、ファイルシステム *filename* に安定を示す印が付けられなかったことを示します。-m オプションを指定して `fsck` を実行すると、この情報を使用して、ファイルシステムの検査が必要かどうか判断されます。

第 33 章

ソフトウェアパッケージで発生する問題の解決 (手順)

この章では、ソフトウェアパッケージをインストールまたは削除するときに発生する問題について説明します。この章には、2つの節があります。「特定のソフトウェアパッケージのインストールエラー」では、パッケージのインストールエラーと管理エラーについて説明します。「一般的なソフトウェアパッケージのインストールに関する問題」では、特定のエラーメッセージを出さない障害について説明します。

この章の内容は次のとおりです。

- 450 ページの「特定のソフトウェアパッケージのインストールエラー」
- 451 ページの「一般的なソフトウェアパッケージのインストールに関する問題」

ソフトウェアパッケージの管理については、『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。

ソフトウェアパッケージのシンボリックリンクに関する問題の解決

以前の Solaris リリースでは、ソフトウェアパッケージを作成するときに、シンボリックリンクのリンク先を `pkgmap` ファイルに指定できませんでした。そのため、`pkgadd` コマンドでパッケージを追加する場合、パッケージまたはパッチ関連のシンボリックリンクには、シンボリックリンクのリンク先ではなくシンボリックリンクのリンク元が使用されていました。したがって、パッケージやパッチパッケージのアップグレードの際にシンボリックリンクのリンク先を別のものに変更する必要があるという問題がありました。

現在のリリースでは、デフォルトの動作では、パッケージのシンボリックリンクのリンク先を変更する必要がある場合、シンボリックリンクのリンク元の代わりにリンク先が `pkgadd` コマンドによって調べられるようになりました。

しかしながら、パッケージによっては pkgadd のこの新しい動作に準拠していない場合があります。

pkgadd シンボリックリンクの新旧動作に対応するために PKG_NONABI_SYMLINKS 環境変数が使用できます。この環境変数が真に設定されていると、pkgadd はシンボリックリンクのリンク元を使用します。

pkgadd コマンドを使ってパッケージを追加する前に管理者がこの変数を設定すれば、新しい動作に対応していないパッケージを以前の動作で処理できます。

pkgadd コマンドを使って既存のパッケージを追加する場合、pkgadd シンボリックリンクの新しい動作が原因でパッケージを追加できないことがあります。その場合には、次のエラーメッセージが表示されます。

```
unable to create symbolic link to <path>
```

この問題のためにパッケージをインストールできない場合は、次の手順に従います。

1. Sun 提供のパッケージの場合は、ご購入先に新動作に対応していないパッケージ名をお知らせください。
2. PKG_NONABI_SYMLINKS 環境変数を設定し、pkgadd コマンドを使ってパッケージを再び追加してください。

```
# PKG_NONABI_SYMLINKS=true  
# export PKG_NONABI_SYMLINKS  
# pkgadd pkg-name
```

特定のソフトウェアパッケージのインストールエラー

```
WARNING: filename <not present on Read Only file system>
```

エラーの原因	問題の解決方法
このエラーメッセージは、パッケージの一部のファイルがインストールできなかったことを示す。このエラーは、通常、pkgadd を使用してパッケージをクライアントにインストールするときに発生する。この場合、pkgadd は、サーバーからマウントしているファイルシステムにパッケージをインストールしようとする。しかし pkgadd は、そのためアクセス権を持っていない	パッケージのインストール中にこの警告メッセージが表示された場合、パッケージをサーバーにもインストールしなければならない。詳細については、『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。

一般的なソフトウェアパッケージのインストールに関する問題

エラーの原因	問題の解決方法
Solaris 2.5 およびその互換バージョンより前に開発された一部のパッケージの追加と削除に関連して、既知の問題が存在する。このようなパッケージを追加または削除すると、ユーザーとの対話中にインストールが失敗するか、ユーザーとの対話のためにプロンプトが出されるが、ユーザーの応答は無視されることがある	次の環境変数を設定して、パッケージを追加し直す。 <code>NONABI_SCRIPTS=TRUE</code>

索引

数字・記号

- 64 ビット Solaris
 - 機能が有効かどうかを決定, 221
 - システムが実行できるかどうかを決定, 220

A

- accept コマンド, 97
- accounting
 - connect, 300
 - reports
 - daily command summary, 300
- acct.h 形式ファイル, 297, 298
- acctcms コマンド, 290, 301
- acctcom コマンド, 297, 298
- acctcon command, 300
- acctcon コマンド, 284, 290
- acctdusg コマンド, 278, 294
- acctdusg コマンド, 299
- acctprc コマンド, 290
- acctwtmp コマンド, 293
- acctwtmp コマンド, 277, 279
- acct ファイル, 281
- active.MMDD ファイル, 285
- activeMMDD ファイル, 300
- active ファイル, 285, 300
- active ファイル, 289
- Admintool, 前提条件, 56
- ASCII ファイル, ファイル内容形式, 50
- at.deny ファイル, 273
- at.deny ファイル, 270
- at.deny ファイル, 説明, 258
- atjobs ディレクトリ, 261
 - 説明, 258
- atq コマンド, 271
- at コマンド, 269, 270, 273
 - l オプション (list), 272
- at コマンド
 - l オプション (list), 271
 - m オプション (mail), 270
- at コマンド
 - m オプション (メール), 271
- at コマンド
 - エラーメッセージ, 273
- at コマンド
 - 概要, 258, 259, 269
- at コマンド
 - 自動スケジュール, 261
- at コマンド
 - 使用制御, 273
- at コマンド
 - 使用制御, 270
- at コマンド
 - 使用制御
 - 概要, 258
 - 使用の拒否, 273
- at ジョブファイル, 269
- at ジョブファイル, 272
- at ジョブファイル
 - 位置, 259
- at ジョブファイル
 - 検査, 271
 - 削除, 272
- at ジョブファイル
 - 作成, 271

at ジョブファイル
作成, 270
at ジョブファイル
説明, 259
at ジョブファイル
登録, 269
at ジョブファイル
表示, 272
at ジョブファイル
表示, 271
待ち行列の表示, 271
auth 機能, 369

B

banner オプション, 85

C

cancel コマンド, 99
catman データベース, 258
chargefee スクリプト, 283, 294
chargefee スクリプト, 279, 283
ckpacct スクリプト, 279, 280, 282
closewtmp コマンド, 290
cmsprev ファイル, 301
configuring
プリンタポート, 47
for IA システム, 47
connect accounting, 300
consadm コマンド, 372
補助コンソールのリストを表示する (方法),
373
補助コンソールを無効にする, 374
補助コンソールを有効にする, 372
システムリブート後, 373
coreadm コマンド, 376
グローバルコアファイルパスを有効にする,
379
コアダンプ構成の表示, 378
コアファイルの管理, 376
コアファイル名パターンの設定, 379
プロセス別コアファイルパスを有効にする,
379
core ファイル, proc ツールを使用した検
査, 380

core ファイル
検索と削除, 238
自動削除, 269
CPU (中央処理装置)
使用時間の長いプロセス, 327
CPU (中央処理装置), 情報の表示, 298
CPU (中央処理装置)
情報の表示, 294, 298, 327
使用時間, 314
cron.allow ファイル, 266, 268
cron.allow ファイル, 267
cron.deny ファイル, 266, 267
crontab コマンド, 267
cron デーモン, 261
-e オプション (edit), 262, 263
-l オプション (list), 264
-r オプション (remove), 265, 266
crontab コマンド
/var/adm の保守, 367
crontab コマンド
アカウントिंगスクリプトの実行, 280,
282
アクセスの制御, 267
エラーメッセージ, 268
概要, 257, 258
使用されるファイル, 261
使用の制御, 266, 267, 268
概要, 258, 266
使用の拒否, 266, 267
特定ユーザーの使用の制限, 268
スケジュール, 261
変更を保存しないで終了, 263
毎日の作業, 258
/crontab ディレクトリ, 260, 261, 262
crontab ファイル, あるかどうかを調べる, 263
crontab ファイル
位置, 260
構文, 261, 262
削除, 265, 266
作成, 262, 263
使用の拒否, 267
説明, 261
表示, 264
編集, 262, 263
cron デーモン, 259
cron デーモン, 261
ctacct.MMDD file, 300
ctacctMMDD ファイル, 290

ctmp file, 300

D

daily accounting

reports

daily command summary, 300

date command, described, 219

date コマンド

アカウントリングデータ, 277, 279

daytacct file, 300

daytacct ファイル, 290, 301

daytacct ファイル, 294

deadfiles ファイル, 237

df コマンド, 337, 338

df コマンド

-F オプション (マウントされていないファイルシステム), 228

-g オプション (statvfs 構造体), 228

df コマンド

--h オプション, 229

-k オプション (キロバイト), 337, 338

df コマンド

-k オプション (キロバイト), 228

-t オプション (合計ブロック), 228, 229

df コマンド

概要, 228, 337

例, 337, 338

df コマンド

例, 228

disable コマンド, 95

disable コマンド、例, 96

disktacct file, 300

disktacct.MMDD ファイル, 290

disktacct ファイル, 279, 290

disktacct ファイル, 278

dispadmin コマンド、概要, 321

displaying

date, 219

host ID, 219

system information

commands for, 219

time, 219

dmesg コマンド, 367

DNLC 機能の向上, 305

dodisk script, files created by, 300

dodisk スクリプト, 277

dodisk スクリプト

概要, 278

dodisk スクリプト

概要, 277, 279

dodisk スクリプト

作成されたファイル, 290

dodisk スクリプト

作成されたファイル, 278, 279

dodisk スクリプト

実行するcrontab エントリ, 282

dodisk スクリプト

注意, 278

dtmp ファイル, 299

dumpadm

クラッシュダンプ情報の表示, 385

クラッシュダンプ情報の変更, 385

システムクラッシュ情報の管理, 383

du コマンド, 233, 235

ディレクトリサイズの表示, 233

E

edquota コマンド

-p オプション (プロトタイプ), 247

-t オプション (期間制限), 253

edquota コマンド、概要, 244

edquota コマンド

概要, 243, 252

各ユーザーのディスク割り当て制限を変更, 253

特定のユーザーのディスク割り当て制限を無効にする, 254, 255

ユーザーディスク割り当て制限の設定, 246, 247

enable コマンド, 95

enable コマンド、例, 96

/etc/acct/holidays ファイル, 294

/etc/acct/holidays ファイル, 282

/etc/cron.d/at.deny ファイル, 270

/etc/cron.d/at.deny ファイル, 273

/etc/cron.d/cron.allow ファイル, 267

/etc/cron.d/cron.allow ファイル, 266, 268

/etc/cron.d/cron.deny ファイル, 266, 267

/etc/init.d/acct ファイル, 281

/etc/init.d/perf ファイル, 355, 357

/etc/inittab ファイル, 196

- /etc/lp/classes/printer-class file, 87
- /etc/lp/default ファイル, 83
- /etc/lp/fd ディレクトリ, 171
- /etc/lp/fd ディレクトリ, 115
- /etc/lp/filter.table ファイル, 115
- /etc/lp/filter.table ファイル
 - 削除されたフィルタ, 117
 - 追加されたフィルタ, 116
- /etc/lp/forms/form-name/describe ファイル, 121
- /etc/lp/forms/form-name ファイル, 122
- /etc/lp/forms ディレクトリ, 120
- /etc/lp/printers ディレクトリ, 162
- /etc/lp/printers ディレクトリ
 - 印刷クライアント, 77
 - プリンタサーバー, 78
- /etc/lp/Systems ファイル, 77
- /etc/lp ディレクトリ, 161
- /etc/motd ファイル, 225
- /etc/printcap ファイル, 162
- /etc/saf/_sactab ファイル, 196
- /etc/syslog.conf ファイル, 368
- /etc/utmpx ファイル, 198
- /etc/vfstab ファイル, 245, 248

F

- fcntl 情報, 317
- fcntl 情報, 317, 318
- fd2log file, 300
- fd2log ファイル, 285, 289
- fee ファイル, 279, 283, 290, 299
- find コマンド, 237
- find コマンド, 232, 235, 238
- fiscrptn ファイル, 301
- fsck コマンド, 258
- fstat 情報, 317
- fstat 情報, 317, 318

G

- getty, 183

H

- hog 係数
 - レポート, 296, 298
- holidays ファイル, 282
- holidays ファイル, 294
- hostid command, 219
- hostid コマンド, 223

I

- igets コマンド, ページフラッシュ, 343
- init プロセス, 196
- init プロセス, 277
- iostat コマンド
 - xtc オプション (拡張), 336
 - 概要, 334
 - 基本情報の表示, 335
 - ディスクの使用状況の表示, 335
- i ノード, 339, 343

K

- kern 機能, 369
- kill コマンド, 317, 320
- klwp 構造体, 310
- kthread 構造体, 310

L

- lastdate file, 300
- lastdate ファイル, 290
- lastlogin コマンド, 290
- lineuse ファイル, 290
- lineuse ファイル, 301
- lock1 ファイル, 290
- lock ファイル, 290
- log file, 300
- log.MMDD ファイル, 300
- loginlog ファイル, 290, 301
- login コマンド, 277
- lpadmin コマンド
 - o banner オプション, 85
- lpadmin コマンド
 - 印字ホイール装着の警告の設定 (方法), 111

- lpadmin コマンド
 - 印字ホイールの装着 (方法), 110
 - 印字ホイールの定義, 108
 - デフォルトプリンタの設定 (方法), 83
 - バナーページを選択可能にする (方法), 84
 - フォームの装着に関する警告の設定 (方法), 123
 - フォームの装着 (方法), 122
 - フォームの取り外し (方法), 122
 - lpadmin コマンド
 - フォームへのプリンタアクセスの制限, 127
 - lpadmin コマンド
 - フォントカートリッジの装着 (方法), 110
 - フォントカートリッジの定義, 108
 - プリンタ記述の追加 (方法), 81
 - プリンタクラスの定義 (方法), 87
 - プリンタの障害回復の設定 (方法), 90
 - プリンタの障害警告の設定 (方法), 88
 - プリンタへのアクセスの制限 (方法), 92
 - プリンタポート特性の調整 (方法), 134
 - LPDEST 環境変数, 83
 - lpfilter コマンド, 115
 - lpsched
 - デーモン (リファレンス), 176
 - ログファイル, 172
 - lpsched デーモン, 169, 171
 - LP 印刷サービス
 - 印字ホイールの確認, 108
 - インタフェースプログラム, 171
 - カスタマイズ, 157
 - 基本機能のチェック, 413, 416
 - 構成ファイル, 161
 - 構造, 160
 - 参照情報, 160
 - 使用されるファイル, 163
 - スケジューラ, 169
 - 定義, 160
 - ディレクトリ, 160
 - デーモン, 163
 - ハングした LP コマンド, 421
 - フォームの追跡 (概要), 119
 - プリンタ特性の定義, 43
 - 問題の解決, 406, 409
 - ログファイル, 407
 - LP コマンド, 421
 - LP プリントサービス, ログファイル, 163
 - ls コマンド, 236
 - ls コマンド, 230, 231, 236
 - ls コマンド (続き)
 - l オプション (バイト数単位のサイズ), 230, 231
 - ls コマンド, -s オプション (ブロック数単位のサイズ), 231
 - ls コマンド
 - s オプション (ブロック数単位のサイズ), 230, 231
 - t オプション (最新ファイル), 236
 - LWP (軽量プロセス)
 - 構造体, 310
 - 情報の表示, 317
 - 定義, 309
 - プロセス, 309, 310
- ## M
- mdb コーティリテイ, 387
 - memory, command for displaying information on, 219
 - messages.*n* ファイル, 366
 - messages ファイル, 364, 368
 - MMDD ファイル, 291
 - monacct スクリプト
 - runacct スクリプト, 280, 289
 - monacct スクリプト
 - 月次コマンドの要約, 295, 296
 - monacct スクリプト
 - 実行スケジュール, 280
 - 実行する crontab エントリ, 282
 - 使用/生成されるファイル, 301
 - MOTD (その日のメッセージ) 機能, 225
 - motd ファイル, 225
 - motd ファイル, 225
- ## N
- nice コマンド, 325, 327
 - nice 番号, 314
 - nice番号, 325
 - nlsadmin コマンド, 200
- ## O
- owtmpx ファイル, 301

P

pacctn ファイル
概要, 290

pacctn ファイル
概要, 279, 299

pacctn ファイル
サイズの監視, 279, 289

pacctn ファイル
表示, 297, 298

panic メッセージ, 366

perf ファイル, 355

pfiles コマンド, 317, 318

pflags コマンド, 317

pldd コマンド, 317

pmadm コマンド
ttymon サービスの追加, 204
ttymon サービスの表示, 205
ttymon サービスを無効にする, 208
ttymon サービスを有効にする, 208

pmadm コマンド, 説明, 196

pmap コマンド, 317

PostScript 以外のプリンタ, 51, 106, 108

PostScript フォント, 128
インストール (方法), 130

PostScript プリンタ, 106
デフォルトの印刷フィルタ, 115
ファイル内容形式, 50
プリンタタイプ, 49
文字セット, 107

prdaily script, files used by, 300

prdaily スクリプト
runacct スクリプト, 289, 301
回線利用状況のレポート, 301
概要, 289
使用されるファイル, 300, 301

.printers ファイル, 設定 (方法), 63

PRINTER 環境変数, 83

priocntl コマンド
-c オプション (スケジューリングクラス指定)
, 323, 324
-e オプション (実行), 323
-i オプション (識別タイプ), 323, 324
-l オプション (スケジューリングクラスの表示), 321, 322
-m オプション (最上位/最下位優先順位)
, 323
-p オプション (優先順位の指定), 323

priocntl コマンド (続き)
-s オプション (優先順位の上限/優先順位の変更), 323, 324
概要, 321

PROCFS (プロセスファイルシステム), 316

proc 構造体, 310, 314

proc ツール, コアファイルの検査, 380

/proc ディレクトリ, 316

prtconf command, 219

prtconf コマンド, 223

psig コマンド, 317

PSR プリンタタイプ, 49

pstack コマンド, 317

ps コマンド, 314, 316
-c オプション (スケジューリングクラス)
, 314
-ecl オプション (グローバル優先順位), 322
-ef オプション (詳細情報), 315, 316
概要, 314
出力レポートのフィールド, 314, 325, 327

PS プリンタタイプ, 49

ptime コマンド, 317

ptree コマンド, 317, 318

pwait コマンド, 317

pwdx コマンド, 317, 318

Q

quotacheck コマンド, 244

quotacheck コマンド, 243, 248

quotaon コマンド, 249

quotaon コマンド, 249

quotas
setting up, 243
削除, 252

quotas ファイル, 243

quotas ファイル, 245, 246

quota コマンド, 243, 249, 250, 253

quot コマンド, 234, 235

R

reboots ファイル, 290, 300

reject コマンド, 97

repquota コマンド, 251

repquota コマンド, 249, 250

residentfonts ファイル, 129
Return 設定、プリンタ, 409
Reverse PostScript プリンタ
印刷方法, 49
ファイル内容形式, 51
プリンタタイプ, 49, 51
rm コマンド, 237, 238
root ファイル, 260
rprt.MMDD ファイル, 280, 301
rpt.MMDD ファイル, 290, 301
RS-232-C
シリアルポートを参照
runacct script, last time executed, 300
runacct スクリプト, 289
monacct スクリプト, 289
prdaily スクリプト, 289, 301
エラー保護, 289, 290
エラーメッセージ, 285
概要, 279
壊れたファイルの修復, 284, 290
壊れたファイルの復元, 284
再起動, 285, 286, 290, 291
実行スケジュール, 280
実行する crontab エントリ, 289
失敗, 285
使用/生成されるファイル, 301
状態, 290
使用または生成されるファイル, 299, 301
診断ファイル, 289
進捗ファイル, 289
ユーザー料金の計算, 283, 294

S

sa1 コマンド, 354
sa2 コマンド, 354, 355
SAC
サービスアクセスコントローラを参照
sacadm コマンド
ttymon ポートモニターの起動, 203
ttymon ポートモニターの削除, 204
ttymon ポートモニターの終了, 203
ttymon ポートモニターの追加, 201
ttymon ポートモニターの表示, 202
ttymon ポートモニターを無効にする, 203
ttymon ポートモニターを有効にする, 203
sacadm コマンド, 説明, 195

sadc コマンド, 354, 355, 357
sadd ファイル, 355
SAF
サービスアクセス機能 (SAF)を参照
sar コマンド, 338, 355
-A オプション (全体のパフォーマンス), 354, 356
-a オプション (ファイルアクセス), 338
sar コマンド, -a オプション (ファイルアクセス)
, 339
sar コマンド
-b オプション (バッファ), 339, 340
-c オプション (システムコール), 340, 341
-d オプション (ブロックデバイス), 341
-e オプション (終了時刻), 356
-f オプション (データを抽出するファイル)
, 356
-g オプション (ページアウト/メモリーの解放), 343
-i オプション (間隔), 356
-k オプション (カーネルメモリー), 344
-m オプション (プロセス間通信), 345
-p オプション (ページイン/ページフォルト)
, 346
-q オプション (待ち行列), 347, 348
-r オプション (未使用メモリー), 348
-s オプション (開始時刻), 356
-u オプション (CPU 使用率), 349
-v オプション (システムテーブル), 351
-w オプション (ボリュームのスワップと切り
替え), 352
-y オプション (端末の動作), 353
オプションのリスト, 356
概要, 338, 355
すべてのオプションの説明, 356
shutacct スクリプト, 279, 280
shutdown コマンド, 280
simple ファイル内容形式, 50
Solaris と lpd をベースにした印刷システム, 43
Solaris プリンタマネージャ
概要, 30
プリンタアクセスを追加, 61
プリンタの新規追加 (方法), 59
プリンタの定義機能 (概要), 45
Spacctn.MMDD ファイル, 290
Spacctn.MMDD ファイル, 299
startup コマンド, 278
statefile ファイル, 285, 290, 300

- statvfs 構造体, 228
- stty 設定
 - カスタマイズ, 140
 - 推奨事項, 420
 - デフォルト, 133, 420
 - 問題の解決, 408
- sys crontab, 355
- syslog.conf ファイル, 368
- syslogd デーモン, 366
- sys ファイル, 260

T

- tacct.MMDD ファイル, 290, 301
- tacct.MMDD ファイル, 285
- tacctn ファイル, 301
- tacctprev ファイル, 301
- tacct ファイル, 285, 290, 301
- terminfo エントリ
 - サポートされていないプリンタのものを追加 (方法), 138
 - 選択可能文字セット, 106
- terminfo エントリ, 追加 (概要), 135
- terminfo エントリ, 必要項目のリスト, 136
- terminfo データベース, 文字セット名, 106
- time, displaying, 219
- /tmp/disktacct.MMDD ファイル, 290
- tmpwtmp ファイル, 290, 300, 301
- /tmp ディレクトリのクリア, 237, 238
- TOTAL REAL-MIN 列 (日次コマンドの要約), 295
- total コマンドの要約, 296
- TranScript フィルタ, 115
- troff フィルタによる PostScript への変換, 154
- ttyadm コマンド, 200
- ttymon コマンド, 277
- ttymon サービス
 - 追加, 204
 - 表示, 205
 - 無効にする, 208
 - 有効にする, 208
- ttymon ポートモニター
 - 起動, 203
 - 機能の概要, 196
 - 削除, 204
 - 終了, 203
 - 図, 197

- ttymon ポートモニター (続き)
 - 追加, 201
 - 発着信両用サービス, 198
 - 表示, 202
 - 無効にする, 203
 - 有効にする, 203
- tty 回線, 使用状況の監視, 292
- tty 回線
 - 使用状況の監視, 277, 293, 301
 - 使用の監視, 277
 - 不良回線の問題の解決, 293
- turnacct switch スクリプト, 279, 290

U

- UFS ファイルシステム
 - 情報の表示, 235
 - ユーザーの所有権, 234
- uname コマンド, 222
- UNIX システム (クラッシュ情報), 382
- user 構造体, 310
 - /usr/adm/messages ファイル, 364
 - /usr/bin/mdb ユーティリティ, 387
 - /usr/lib/acct/startup コマンド, 278
 - /usr/lib/lp/model ディレクトリ, 171
 - /usr/lib/lp/postscript ディレクトリ, 171
 - /usr/lib/lp/postscript ディレクトリ, 114
 - /usr/lib/lp ディレクトリ, 115
 - /usr/proc/bin ディレクトリ, 316, 317
 - utmp2wtmp コマンド, 290

V

- /var/adm/acct/fiscal ディレクトリ, 300
- /var/adm/acct/nite/active.MMDD ファイル, 289
- /var/adm/acct/nite/active.MMDD ファイル, 300
- /var/adm/acct/nite/active ファイル, 285, 289, 300
- /var/adm/acct/nite/cms ファイル, 290
- /var/adm/acct/nite/cms ファイル, 300
- /var/adm/acct/nite/ctacct.MMDD ファイル, 290, 300
- /var/adm/acct/nite/ctmp file, 300

/var/adm/acct/nite/daycms file, 300
 /var/adm/acct/nite/daycms ファイル, 290, 301
 /var/adm/acct/nite/daytacct file, 300
 /var/adm/acct/nite/daytacct ファイル, 294
 /var/adm/acct/nite/daytacct ファイル, 290, 301
 /var/adm/acct/nite/disktacct file, 300
 /var/adm/acct/nite/disktacct.MMDD ファイル, 290
 /var/adm/acct/nite/disktacct ファイル, 278, 279, 290
 /var/adm/acct/nite/fd2log file, 300
 /var/adm/acct/nite/fd2log ファイル, 285, 289
 /var/adm/acct/nite/lastdate file, 300
 /var/adm/acct/nite/lastdate ファイル, 290
 /var/adm/acct/nite/lineuse file, 300
 /var/adm/acct/nite/lineuse ファイル, 290, 301
 /var/adm/acct/nite/lock file, 300
 /var/adm/acct/nite/lock1 ファイル, 290
 /var/adm/acct/nite/lock ファイル, 285, 290
 /var/adm/acct/nite/log file, 300
 /var/adm/acct/nite/log.MMDD ファイル, 300
 /var/adm/acct/nite/owtmpx ファイル, 301
 /var/adm/acct/nite/reboots ファイル, 290, 300
 /var/adm/acct/nite/statefile ファイル, 285
 /var/adm/acct/nite/statefile ファイル, 290, 300
 /var/adm/acct/nite/tmpwtmp ファイル, 290, 300, 301
 /var/adm/acct/nite/wtmp.MMDD file, 290
 /var/adm/acct/nite/wtmp.MMDD ファイル, 300
 /var/adm/acct/nite/wtmperror.MMDD ファイル, 300
 /var/adm/acct/nite/wtmperror ファイル, 300
 /var/adm/acct/nite ディレクトリ, 300
 /var/adm/acct/sum/cmsprev ファイル, 301
 /var/adm/acct/sum/cms ファイル, 290
 /var/adm/acct/sum/cms ファイル, 301
 /var/adm/acct/sum/daycms ファイル, 301
 /var/adm/acct/sum/daycms ファイル, 290
 /var/adm/acct/sum/loginlog ファイル, 290, 301
 /var/adm/acct/sum/rprpt.MMDD ファイル, 301
 /var/adm/acct/sum/rprptMMDD ファイル, 280
 /var/adm/acct/sum/rpt.MMDD ファイル, 290
 /var/adm/acct/sum/tacct.MMDD ファイル, 285, 290, 301
 /var/adm/acct/sum/tacctprev ファイル, 285, 301
 /var/adm/acct/sum/tacct ファイル, 290
 /var/adm/acct/sum/tacct ファイル, 285, 301
 /var/adm/acct/sum ディレクトリ, 280, 300, 301
 /var/adm/acct ディレクトリ, 300
 /var/adm/dtmp ファイル, 299
 /var/adm/fee ファイル, 299
 /var/adm/fee ファイル, 279, 283, 290
 /var/adm/messages.n ファイル, 366
 /var/adm/messages ファイル, 364, 368
 /var/adm/sa/sadd ファイル, 355
 /var/adm/Spacctn.MMDD ファイル, 299
 /var/adm/Spacctn.MMDD ファイル, 290
 /var/adm ディレクトリ
 raw アカウンティングデータ, 278
 サイズの制御, 237
 説明, 299
 /var/lp/logs/lpsched ファイル, 171
 /var/lp/logs/lpsched ファイル, 407
 /var/lp/logs ディレクトリ, 172
 /var/spool/cron/atjobs ディレクトリ, 261
 /var/spool/cron/atjobs ディレクトリ, 258, 259
 /var/spool/cron/crontabs/adm ファイル, 260
 /var/spool/cron/crontabs/lp ファイル, 260
 /var/spool/cron/crontabs/lp ファイル, 172
 /var/spool/cron/crontabs/root ファイル, 260
 /var/spool/cron/crontabs/root ファイル, 278

/var/spool/cron/crontabs/sys
crontab, 355
/var/spool/cron/crontabs/sys ファイル, 260
/var/spool/cron/crontab ディレクトリ, 260,
261
/var/spool/lp/requests ディレクトリ, 164
/var/spool/lp/tmp ディレクトリ, 164
/var/spool/lp ディレクトリ, 44, 167
/var/spool ディレクトリ, 237, 238
/var/tmp/deadfiles ファイル, 237
/var/tmp ディレクトリ, 237, 238
/var パーティション, 44
vfstab ファイルとディスク割り当て制限, 245,
246, 248
vmstat コマンド
-c オプション (キャッシュフラッシュ), 333
-i オプション (割り込み), 334
-s オプション (システムイベント), 332
-S オプション (スワッピング), 333
概要, 330
仮想メモリーの表示, 331
レポートのフィールド, 330

W

Watchdog reset! メッセージ, 366
wtmp.MMDD ファイル, 290, 300
wtmperror.MMDD ファイル, 300
wtmperror ファイル, 300
wtmpfix コマンド, 284, 290, 300
wtmpx ファイル
概要, 284, 290
壊れたファイルの修復, 284, 290
シャットダウン, 280
日次レポート, 293

あ

アカウントティング, 286, 301
raw データ, 278
維持, 284, 286
概要, 276
課金ユーザー, 283, 294
起動, 282
壊れたファイルの修復
wtmpx ファイル, 284, 290

アカウントティング (続き)
自動実行のための設定 (方法), 281
種類, 283
使用されるファイル, 299
使用不可, 287
接続, 277, 279, 290, 293, 294
停止, 286
ディスク, 277, 278, 279, 294
日次, 278, 298, 301
手順の要約, 280
レポート, 292
ファイル, 301
プロセス, 277, 279, 293, 294
ユーザーへの課金, 279, 283
ユーザー料金の計算, 279, 283
レポート, 292, 293, 295, 296, 297, 298
概要, 292
合計コマンド要約 (月次), 301
日次コマンドの概要, 290
日次コマンド要約, 301
日次利用状況レポート, 294
日次レポート (tty line utilization), 292

アクセス

フォーム (方法)
プリンタの制限, 127
ユーザーの制限, 126
プリンタ
削除 (方法), 76
アスタリスク (*), crontab ファイル, 261
アダプタボード (シリアルポート), 183
アドレス空間マップ, 317
アプリケーションスレッド, 309, 311

い

一時ディレクトリ, 235, 237, 238
位置揃えパターン
印刷, 123
定義, 156
保護, 120
印刷
状態メッセージ, 424
スプーリングディレクトリ, 167
停止 (概要), 95
特殊モード, 145
バナーページ, 83
プリンタの受け付けと拒否の設定 (概要), 97

- 印刷 (続き)
 - ユーザー料金の計算, 283
 - リモート, 175
 - ローカル (図), 172
- 印刷クライアント
 - 構成の確認, 414
 - ジョブの解放, 424
 - 定義, 42
 - プリンタへのアクセスの削除 (方法), 76
- 印刷構成
 - Solaris と lpd をベースにした印刷システムの使用, 43
 - 集中化, 42
- 印刷スケジューラ
 - LP システムファイルの更新, 169
 - 起動 (方法), 81
 - 再起動 (方法), 80
 - 停止 (方法), 80
 - 動作していない場合, 413
- 印刷デーモン, 163
- 印刷フィルタ
 - PostScript, 115
 - TranScript, 115
 - troff から PostScript への変換, 154
 - 印刷要求の保留, 146
 - オプションのキーワード, 150
 - オプションを定義するテンプレート, 150
 - 管理 (概要), 114
 - 高速 (概要), 129
 - 削除 (概要), 114
 - 削除 (方法), 116
 - 作成 (概要), 114, 144
 - 作成 (方法), 153
 - 種類, 144
 - 情報の表示 (方法), 117
 - ダウンロード (概要), 129
 - 追加 (概要), 114
 - 追加 (方法), 116
 - 定義, 114, 171
 - 定義の作成, 147
 - 低速 (概要), 129
 - 特殊モードの処理, 145
 - 特性, 151
 - バイパス, 418, 419
 - ファイル内容形式の変換, 114
 - ファイルの変換, 145
 - 復元 (概要), 114
 - プリンタ障害からの回復, 90
- 印刷フィルタ (続き)
 - プリンタ障害からの回復 (概要), 90
 - 変更 (概要), 114
 - 要件, 146
- 印刷フォーム
 - フォームを参照
- 印刷待ち行列, ログ, 164
- 印刷待ち行列用のディスク容量, 44
- 印刷要求
 - ID, 83, 94, 95, 99
 - ID (方法), 100, 103
 - 印刷要求の移動, 101
 - 受け付け (方法), 98
 - 拒否 (方法), 77, 98
 - 状態のチェック (方法), 94
 - スケジューリング, 170
 - 他のプリンタへの移動 (概要), 101
 - 他のプリンタへの移動 (方法), 101
 - 取り消し (概要), 99
 - 取り消し (方法), 99
 - 特定ユーザー, 100
 - プリンタを使用不可にする, 96
 - 保留状態 (概要), 103
 - 待ち行列の先頭に移動 (概要), 103
 - 優先順位の変更 (概要), 94, 103
 - 優先順位の変更 (方法), 103
 - ログ, 165
 - ログファイルから消去, 172
- 印刷要求の移動 (概要), 101
- 印刷要求の移動 (方法), 101
- 印刷要求の受け付け (概要), 97
- 印刷要求の受け付け (方法), 98
- 印刷要求の拒否 (概要), 97
- 印刷要求の拒否 (方法), 77, 98
- 印刷要求の優先順位, 94, 103
- 印字ホイール, 107
 - 確認, 108
 - 装着 (方法), 109
 - 装着を促す警告, 111
 - 定義 (方法), 108
 - 取り外し (方法), 109
 - 命名, 108
- インストール
 - PostScript フォント (方法), 130
 - ホスト常駐 PostScript フォント (方法), 131
- インタフェースプログラム (プリンタ)
 - カスタマイズ (概要), 139
 - カスタマイズ (方法), 142

インタフェースプログラム (プリンタ) (続き)
標準, 143

え

エラーメッセージ
at コマンド, 273
crontab コマンド, 268
runacct スクリプト, 285
格納場所の指定, 366, 368
クラッシュメッセージ, 366, 367
ソース, 368
優先順位, 369
ログのカスタマイズ, 368
ログファイル, 364, 366

お

オペレーティングシステム, 情報の表示, 222

か

カーネルスレッド
構造体, 310, 314
スケジューリング, 314
回線使用状況の監視, 277, 292, 293, 301
回線使用の監視, 277
回線制御, 197
書き込みシステムコール
統計情報, 296, 298
課金ユーザー, 283, 294
確認
ディスク割り当て制限, 243, 249, 253
カスタマイズ
LP 印刷サービス, 157
stty モード, 140
システムメッセージ記録, 368
システムメッセージ記録 (方法), 369
終了コード、プリンタ, 140
プリンタインタフェースプログラム (概要)
, 139
プリンタインタフェースプログラム (方法)
, 142
カスタマサービス, クラッシュ情報の送信, 364

環境変数

LPDEST, 83
PRINTER, 83

管理

印刷フィルタ (概要), 114
フォーム, 118, 127
フォント (概要), 128
文字セット, 106

き

期間

弱い制限, 252, 253

起動

Solaris プリンタマネージャ (方法), 56
印刷スケジューラ (方法), 81
共有メモリー, プロセス仮想メモリー, 311
許可リスト

フォームへのプリンタアクセス, 127
フォームへのユーザーアクセス, 126
プリンタへのユーザーアクセス (概要), 92

拒否リスト

フォームへのプリンタアクセス, 127
フォームへのユーザーアクセス, 126
プリンタへのユーザーアクセス (概要), 92
キロバイト, ファイルシステムのディスク使用, 337

く

クラス (プリンタ), 86
lpadmin コマンドによる定義 (方法), 87
使用可能または使用不可にできない, 96
状態のチェック, 94

クラッシュ, 368, 394

カスタマサービス, 364, 383
クラッシュ後のレポートの失敗, 389
クラッシュダンプ情報の保存, 382
クラッシュダンプの検査, 387
クラッシュダンプファイルの削除, 239
生成されたシステム情報の表示, 366, 387
他のシステム情報の保存, 367
手順, 364, 394
クラッシュダンプ
保存を無効にする, 388
保存を有効にする, 388

- クラッシュダンプ構成
 - dumpadm を使用した表示, 385
 - dumpadm を使用した変更, 385
 - クラッシュダンプ情報の保存, 383
 - クラッシュダンプディレクトリ, 一杯になったクラッシュダンプディレクトリからの回復, 387
 - グローバルコアファイルの設定, coreadm を使用した設定, 376
 - グローバル優先順位
 - 定義, 321
 - 表示, 322
- け
- 警告
 - フォームの装着 (概要), 120
 - フォームの装着 (方法), 123
 - プリンタの障害 (方法), 88
 - 文字セットの装着, 111
 - 警告メッセージ, 450
 - 警告メッセージの優先順位 (syslogd の), 369
 - 月次コマンドの要約, 296
 - 検査
 - at ジョブ, 271
 - crontab ファイルの存在, 263
 - 検索
 - サイズ制限を超えるファイル, 232
 - 削除、古い/使用されていないファイル, 235, 239, 263, 271
 - 大規模ファイル, 231
 - 検索パス, 設定ファイル, 400
- こ
- コアダンプ構成, coreadm を使用した表示, 378
 - コアファイル, coreadm を使用した管理, 376
 - コアファイルの検査, proc ツールを使用した, 380
 - コアファイル名パターン, coreadm を使用した設定, 377
 - 合計コマンド要約, 301
 - 構造体, statvfs, 228
 - 高速印刷フィルタ (概要), 129
 - 構文 (crontab ファイル), 261, 262
 - コマンド, 使用を監視, 300
 - 「コマンドが見つかりません」というメッセージ, 399
 - コンソール
 - 補助
 - システムリブート後も有効にする, 373
- さ
- サービスアクセス機能 (SAF)
 - 概要, 184, 194
 - 関連ファイル (表), 208
 - 関連プログラム (表), 194
 - 使用, 194
 - 使用時期, 184
 - 制御されるサービス, 211
 - 設定, 184
 - 説明, 184
 - サービスアクセスコントローラ, 195
 - 最下位, nice 番号, 325
 - 再開, プロセス, 317
 - 最下位, 優先順位, 323
 - 再起動
 - runacct スクリプト, 285, 286, 290, 291
 - 印刷スケジューラ (方法), 80
 - 最終ログインレポート, 297
 - 最上位
 - nice 番号, 325
 - 優先順位, 323
 - サイズ
 - ディレクトリ, 233, 235
 - du コマンド, 233
 - ファイル, 230, 231, 233, 235
 - 最大
 - 最大サイズを超えるファイルの検索, 232
 - 最大値, pacct ファイルのサイズ, 277
 - 削除
 - at ジョブ, 272
 - core ファイル, 238
 - crontab ファイル, 265, 266
 - 一時ファイル, 237, 238
 - 印刷フィルタ (方法), 116
 - クラッシュダンプファイル, 239
 - フォーム (方法), 121
 - プリンタへのアクセス (方法), 76
 - 古いまたは使用されていないファイル, 235, 239, 258, 263
 - ログファイル, 263

作成

- at ジョブ, 270, 271
 - crontab ファイル, 262, 263
 - 印刷フィルタ (概要), 144
 - 印刷フィルタ (方法), 153
 - フォーム, 155
 - フォーム定義 (方法), 157
- サポートされていないプリンタ, 52, 135

し

- シェル, 295
- 時間
 - CPU 時間が大量に増えているプロセス, 327
 - CPU 使用, 294, 314, 327
 - 変更, 224
- 時刻, 表示, 223
- システムイベント, スケジューリング, 267
- システムクラッシュ情報, dumpadm を使用した管理, 383
- システムクラッシュ情報の管理, dumpadm を使用した, 383
- システム資源
 - アカウントिंग
 - 概要, 276
 - 概要, 309
 - 監視, 270, 394
 - アカウントिंग, 286
 - アカウントングシステム, 301
 - クラッシュ, 368
 - 自動, 270
 - ディスク割り当て制限, 251
 - プリンタサーバーの割り当て, 43
- システム終了状態, 298
- システムタスク
 - 自動実行のスケジューリング, 257
 - スケジューリング
 - 1 回だけのタスク, 258, 259, 269, 270, 273
 - 繰り返し行われるタスク, 268
- システムタスク実行の自動化, 258
- システムタスクの自動実行
 - 1 つのタスク, 269, 270, 273
 - 繰り返し行われるタスク, 268
- システム動作
 - データの自動収集, 354, 355, 357
- システム動作データの自動収集, 354, 355, 357
- システム動作データの自動報告, 354

- システム動作の自動報告, 355
- システムの動作
 - システム動作の追跡リスト, 311
 - データの手動収集, 338, 355
- システムメッセージ
 - 格納場所の指定, 366
 - ログのカスタマイズ (方法), 369
- システムメッセージ記録 (カスタマイズ), 368
- 自動システムイベント実行, イベントの繰り返し, 267
- シャットダウン
 - 監視, 279, 280, 293
- 集中化, 印刷構成, 42
- 修復
 - 壊れた wttmpx ファイル, 284
- 終了
 - crontab コマンドの変更を保存しない
で, 263
 - プログラムの強制終了, 394
- 終了コード (プリンタインタフェース), 140
- 終了コード (プリンタインタフェース)
表, 140
 - 標準規格, 140
- 障害回復 (プリンタ), 46
- 障害通知 (プリンタ)
 - lpadmin コマンドによる設定 (概要), 87
 - lpadmin コマンドによる設定 (方法), 88
 - Solaris プリンタマネージャ による設定, 46
 - 警告値, 88
- 使用可能, プリンタ (方法), 96
- 状態
 - 印刷要求のチェック, 94
 - プリンタの取得 (方法), 79
- 状態、(runacct スクリプト), 290
- 使用不可
 - システムアカウントिंग, 287
 - プリンタ (方法), 77, 96
- 所有権, 変更する理由, 402
- シリアルプリンタをシステムに追加, 47
- シリアルポート
 - アダプタボード, 183
 - 定義, 182
- シリアルポートツール, 端末とモデム, 184

す

- スーパーユーザー (ルート) パスワード、忘れた IA, 392
- SPARC, 391
- スケジューリング
 - 1 回だけのシステムタスク, 258, 259, 269, 270, 273
 - 繰り返し実行するシステムタスク, 257, 268
- スケジューリングクラス, 321
 - 指定, 323
 - 情報の表示, 314, 321, 322
 - デフォルト, 323
 - 変更, 324, 325
 - 優先順位の変更, 323, 325
 - 優先順位のレベル, 321, 323
- スプーリングディレクトリ, 印刷サービス, 167

せ

制御

- at コマンドの使用, 258
- at コマンドの使用, 270, 273
- crontab コマンドの使用, 267, 268
 - 概要, 258
- フォームへのプリンタアクセス (方法), 127
- フォームへのユーザーアクセス (方法), 126
- プロセス, 319
- 制御がきかないプロセス, 327
- 制限
 - フォームへのプリンタアクセス (方法), 127
 - フォームへのユーザーアクセス (方法), 126
- 整合性の確認, 248
- セキュリティ
 - at コマンド, 270
 - crontab コマンド, 267
- 接続アカウンティング, 277, 279, 293, 294
- 設定
 - coreadm コマンドを使用したコアファイル名パターン, 379
 - 印刷ホイールの装着を促す警告 (方法), 111
 - 選択可能文字セットの別名 (方法), 112
 - 選択可能文字セットの別名 (例), 113
 - フォントカートリッジの装着を促す警告 (方法), 111
 - 文字セット (例), 113
- 選択可能文字セット, 106, 107, 112, 113
- 選択可能文字セットの別名, 107, 112, 113

そ

装着

- 印刷ホイール (方法), 109
- フォーム (概要), 119
- フォーム (方法), 122
- フォントカートリッジ (方法), 109
- その日のメッセージ (MOTD) 機能, 225
- ソフトウェアパッケージ, インストールの問題の解決, 449

た

- 大規模ファイル, 231
- タイムシェアリングプロセス
 - スケジューリングパラメータの変更, 323
 - 優先順位, 321, 323, 325
 - 変更, 325
- ダウンロード
 - フォント, 419
 - フォント (概要), 128
 - ホスト常駐フォント (概要), 129
- ダウンロードされた PostScript フォント, インストール (方法), 130
- タブ設定, 408
- 端末
 - 英数字, 182
 - 回線使用状況の監視, 277, 292, 293, 301
 - 回線使用の監視, 277
 - 管理用ツール, 184
 - シリアルポートツールの概要, 186
 - シリアルポートツールの項目の説明, 186
 - 設定 (方法), 188
 - 端末間の相違, 182
 - 定義, 182
 - 不良回線の問題の解決, 293
 - プロセス制御, 314
- 端末特性, 140

ち

- 着信専用サービス, 182
- チューニング, 日次コマンドの要約, 295
- 直接印刷, 50

つ

追加

- terminfo エントリ (概要), 135
- 印刷フィルタ (方法), 116
- フォーム (方法), 120
- プリンタ記述 (方法), 81
- プリンタ、新規 (方法)
 - Solaris プリンタマネージャ, 59
- リモートプリンタへのアクセス, 61
- ローカルプリンタまたは接続プリンタ (方法), 59

追跡フラグ, 317

ツール

- システムパフォーマンスの監視, 312
- プロセス, 317
- プロセス情報の表示, 316

て

定義

- 印刷ホイール (方法), 108
- フォントカートリッジ (方法), 108
- 定型作業の自動実行 (概要), 257
- 定型的なシステムイベント, 267
- 定型的なシステムタスクのスケジューリング, 267

停止

- 一時プロセス, 317
- 印刷スケジューラ (方法), 80
- システムアカウントिंग, 286
- プリンタ (概要), 95

ディスクアカウントिंग, 277, 278, 279, 294

ディスクドライブ

- サイズ制限を超えるファイルの検索, 232
- 情報の表示, 228, 230, 231
 - ディスク領域の解放, 337
- 大規模ファイルの検索, 231
- 古いまたは使用されていないファイルの検索と削除, 235, 263
- 容量, 228

ディスク容量

- 最適化, 227
- 情報の表示
 - ディレクトリサイズ, 233, 235
 - ファイルサイズ, 233
 - マウント先, 337
 - ユーザー 1 人あたりのディスク容量, 235

情報の表示 (続き)

- ユーザーの所有権, 234
- 大規模ファイルの検索, 231
- ファイルシステムの使用, 228
- 古い/使用されていないファイルの検索と削除, 239
- ディスク容量の最適化, 227
- ディスク容量のユーザーの所有権, 235
- ディスク領域
 - 空き容量, 337
 - 情報の表示
 - df コマンド, 337
 - ファイルシステムの使用, 337
- ディスク割り当て, 初期設定, 244
- ディスク割り当て制限, 251
 - 概要, 243
 - 確認, 243, 249, 253
 - 各ユーザー用に変更, 253
 - 管理, 251
 - 情報の表示, 243, 249, 250
 - 初期設定, 248
 - 整合性の確認, 248
 - チェック, 249
 - 特定のユーザーについて無効にする, 254, 255
 - ファイルシステムの確認, 250, 251
 - ファイルシステムの構成, 245, 246
 - 複数ユーザー用のプロトタイプ, 247
 - 変更, 252
 - 有効, 243
 - 有効にする, 249
 - ユーザー, 246, 247, 250, 252, 253, 254, 255
 - ユーザーディスク割り当て制限の超過の確認, 250
 - 要件, 244
 - 弱い期間制限, 252, 253
- ディスク割り当て制限の初期設定, 248
- ディスク割り当て制限を自動的に有効にする, 243
- ディスク割り当ての初期設定, 244
- 低速印刷フィルタ, 129
- ディレクトリ
 - 一時ディレクトリのクリア, 235, 237, 238
 - サイズ, 233, 235
 - 情報の表示, 230, 231, 233, 235
 - プロセスの現在の作業ディレクトリ, 317
- デーモン
 - lpsched, 169, 171

デーモン (続き)
 lpsched (リファレンス), 176
 印刷, 163
テクニカルサポート
 クラッシュ情報の送信, 364
 クラッシュダンプの分析, 383
デフォルト
 cron.deny ファイル, 266
 crontab ファイル, 260
 nice 番号, 325
 スケジューリングクラス, 323
 その日のメッセージ, 225
 弱い期間制限, 252, 253
デフォルトプリンタ
 lpadmin コマンドによる設定 (概要), 82
 lpadmin コマンドによる設定 (方法), 83
 Solaris プリンタマネージャによる設定, 46
電源異常からの回復, 293
電源を切ってまた入れる, 394
転送文字
 読み取りおよび書き込みシステムコー
 ル, 296, 298
テンプレート (印刷フィルタ), 150

と

取り消し
 印刷要求 (方法), 99
 特定ユーザー, 100
 プリンタを使用不可にする, 96
取り外し
 印字ホイール (方法), 109
 フォーム (方法), 122
 フォントカートリッジ (方法), 109

に

日時
 変更, 224
日次アカウントティング, 278, 301
 手順の要約, 278, 280
 レポート, 292, 293, 295, 297, 298
 概要, 292
 日次コマンド要約, 290, 301
 日次利用状況レポート, 294
 日次レポート (tty 回線利用状況), 292

ね

ネットワーク, アクセスで発生する問題の把握, 402
ネットワークアクセスで発生する問題の把握, 402
ネットワークプリンタ
 設定用作業マップ, 54
 追加, 63
 追加 (方法)
 lp コマンドによる, 69
 Vendor提供ツール, 68
 定義, 63

は

ハードディスク, プリンタサーバーに推奨, 45
発信専用サービス, 182
発信信両用サービス, 182, 198
バナーページ
 印刷しない理由, 84
 出力が正しくない場合の問題の解決, 407
 設定, 46
 lpadmin コマンドによる, 83
 選択可能にする (方法), 84
 lpadmin コマンドにより, 84
 本文が印刷されない, 418
 抑制する (方法), 85
 --o banner オプションによる, 85
パフォーマンス
 監視ツール, 312
 システム動作の監視, 311, 338, 354
 追跡される動作, 311
 動作データの自動収集, 354, 355, 357
 動作データの手動収集, 338, 355
 ファイルアクセス, 338, 339
 プロセス管理, 309, 317, 325
 レポート, 338
パラレルプリンタ, 47
パリティビット, プリンタ, 408

ひ

日付, 表示, 223
表示
 acct.h 形式ファイル, 297, 298
 at ジョブ, 271, 272

表示 (続き)

- at 待ち行列, 271
- coreadm を使用したコアダンプ構成, 378
- crontab ファイル, 264
- dumpadm を使用したクラッシュダンプ構成, 385
- iostat を使用したディスクの使用状況, 335
- LWP 情報, 317
- pacctn ファイル, 297, 298
- vmstat を使用した仮想メモリー, 331
- オペレーティングシステム情報, 222
- クラッシュ情報, 366, 387
- 時刻, 223
- システム情報, 224
- システム動作情報, 338, 355
- 実行中のプロセス, 316
- スケジューリングクラス情報, 314, 321, 322
- ディスク割り当て制限情報, 243, 249, 250
- ディレクトリ情報, 230, 231, 233, 235
- 日付, 223
- ファイルシステム情報, 228, 337
- ファイル情報, 230, 231, 233, 235, 236
- ファイルのサイズ, 230
- ブートメッセージ, 367
- フォームの状態 (方法), 125
- プロセス, 315
- プロセス情報 (方法), 318
- ホスト ID, 223
- マウントされていないファイルシステムのリスト, 228
- 優先順位情報, 314, 322
- リンクされたライブラリ, 317
- 標準プリンタインタフェースプログラム, 143
- ピン構成 (ヌルモデムケーブル用), 411

ふ

ファイル

- fstat 情報と fcntl 情報の表示, 318
- fstat と fcntl の情報表示, 317
- fstat と fcntl の情報表示, 317
- LP 印刷サービスが使用する, 163
- アカウントティング, 299, 301
- アクセス操作の検査, 338, 339
- 検索パスの設定, 400

ファイル (続き)

- 壊れたファイルの修復
 - wtmpx ファイル, 290
- 壊れたファイルの復元, 284, 285
- サイズ, 230, 231, 233, 235
- サイズ制限を超えるファイルの検索, 232
- サイズの表示, 230
- 使用の監視, 277, 294
- 情報の表示, 230, 231, 236
 - サイズ, 233, 235
- 使用率の監視, 278
- 古い/使用されていないファイルの検索と削除, 235, 239, 263, 271
- 古い/使用されていないファイルの削除, 258
- 古い/使用されていないものの検索と削除, 271
- マウントポイント, 232
- ファイルシステム
 - 情報の表示, 228, 337
 - ディスク容量の使用, 228
 - ディスク領域の使用, 337
 - 復元, 283, 294
 - マウントされたファイルシステムの statvfs 構造体, 228
 - マウントされていない, 228
 - マウントポイント, 337
- ファイル内容形式, 50
 - Admintool のメニュー, 50
 - PostScript, 50
 - PostScript 以外のプリンタ, 51
 - simple, 50
 - Solaris プリンタマネージャで設定, 46
 - 印刷フィルタによる変換, 114, 145
 - 共通プリンタ, 51
 - 出力が正しくない場合の問題の解決, 407
- フィルタ, 50
 - ダウンロード, 129
 - ダウンロード (概要), 129
 - なしに印刷, 50
- ブート
 - sadc コマンドの実行, 355
 - 生成されるメッセージの表示, 367
- フォーム
 - アクセスの制御 (概要), 120
 - 削除 (概要), 119
 - 削除 (方法), 121
 - 作成, 155
 - 状態の表示 (方法), 125

- フォーム (続き)
 - 情報の表示 (方法), 125
 - 装着 (概要), 119
 - 装着したフォームの追跡 (概要), 119
 - 装着に関する警告 (概要), 120
 - 装着に関する警告を設定 (方法), 123
 - 装着 (方法), 122
 - 属性の表示 (概要), 120
 - 追加 (概要), 119
 - 追加 (方法), 120
 - 定義、作成 (方法), 157
 - デフォルト値, 155
 - 取り外し (方法), 122
 - プリンタアクセスの制限 (方法), 127
 - 変更 (概要), 119
 - ユーザーアクセスの許可 (方法), 126
 - ユーザーアクセスの拒否 (方法), 126
 - ユーザーアクセスの制限 (方法), 126
 - 用紙 (給紙と取り外し), 122
- フォント
 - PostScript, 128
 - インストール (方法), 131
 - 種類, 106
 - 常時ダウンロード, 128
 - ダウンロード, 419
 - ダウンロード (概要), 128
 - ダウンロードされた PostScript, 130
 - プリンタ常駐, 128
 - プリンタ常駐 (概要), 129
 - ホスト常駐, 128
 - ホスト常駐 (概要), 129, 130
- フォントカートリッジ, 107
 - 装着 (方法), 109
 - 装着を促す警告, 111
 - 定義 (方法), 108
 - 取り外し (方法), 109
 - 命名, 108
- 復元, コマンドの使用, 396
- プライベートインタフェース, 161
- プリンタ
 - PostScript, 106, 107
 - PostScript 以外, 106, 144
 - stty 設定, 140, 420
 - stty 設定値, 420
 - 印刷要求の受け付け (概要), 97
 - 印刷要求の拒否 (概要), 97
 - 印字ホイール, 106
 - インタフェースプログラム, 139, 140, 143
- プリンタ (続き)
 - インタフェースプログラムのカスタマイズ (概要), 139
 - インタフェースプログラムのカスタマイズ (方法), 142
 - 記述, 追加 (方法), 81
 - 機能の制御 (概要), 97
 - 許可リスト (概要), 92
 - 拒否リスト (概要), 92
 - クライアントアクセスの削除 (概要), 76
 - クラス, 86
 - サポートされていないものを追加 (概要), 135
 - 障害, 141, 145, 423
 - 障害回復 (概要), 90
 - 障害警告 (方法), 88
 - 障害通知, 87
 - 使用可能 (概要), 95
 - 使用可能 (方法), 96
 - 状態, 410
 - 状態 (方法)
 - チェック, 79, 94
 - 使用不可 (概要), 95
 - 使用不可 (方法), 77, 96
 - 新規追加 (方法)
 - Solaris プリンタマネージャによる, 59
 - 設定, 53, 56, 133, 408, 420
 - Return, 409
 - タブ, 408
 - パリティ, 408
 - 設定値
 - stty, 420
 - デフォルト, 83
 - ネットワーク, 54, 63
 - ネットワークプリンタの追加 (方法)
 - lp コマンドによる, 69
 - Vendor提供ツールによる, 68
 - パリティビット, 408
 - ファイル内容形式, 418
 - フィルタの影響, 418
 - フォームへのアクセス (例), 121
 - フォントカートリッジ, 106
 - プリンタサーバーから削除 (方法), 76
 - プリンタサーバーからの削除 (方法), 78
 - ボーレート設定, 408
 - 問題の解決, 407, 409, 410, 413, 418, 421, 422, 424
 - 出力が正しくない場合, 417

- 問題の解決 (続き)
 - プリンタタイプが正しくない場合, 417
 - ユーザーアクセスの許可 (方法), 92
 - ユーザーアクセスの拒否 (方法), 92
 - リモート, 175
 - ローカル, 42, 54, 172
 - ローカルまたは接続の追加 (方法)
 - Solaris プリンタマネージャによる追加, 59
- プリンタインタフェースプログラム, 171
- プリンタ記述
 - lpadmin コマンドによる設定 (概要), 82
 - lpadmin コマンドによる追加 (方法), 81
- プリンタクラス, 86
 - lpadmin コマンドによる定義 (方法), 87
 - 使用可能または使用不可にできない, 96
 - 状態のチェック, 94
 - 設定, 46
- プリンタ、警告
 - 警告を参照
- プリンタサーバー
 - アクセスのチェック, 416
 - 構成の確認, 414
 - システム資源の要求, 43
 - 接続の確認, 414
 - 設定 (概要), 59
 - 定義, 42, 59
 - ハードディスクの要件, 45
 - 必要なスプーリング空間, 44
 - プリンタの削除 (方法), 78
- プリンタサーバーに必要なスプーリング空間, 44
- プリンタ常駐フォント (概要), 129
- プリンタ情報, Solaris 2.5.1 からの変換, 73
- プリンタタイプ, 設定の問題の解決, 417
- プリンタの出力先, Solaris プリンタマネージャによる設定, 46
- プリンタの状態, 79
- プリンタのタイプ
 - Solaris プリンタマネージャによる設定, 46
 - terminfo データベースにない場合, 49
 - terminfo データベースの定義, 163
- プリンタポート
 - configuring, 47
 - シリアル, 47
 - 特性、調整 (概要), 133
 - 特性の調整 (方法), 134
 - パラレル, 47
- プリンタポート (続き)
 - 複数ポートを有効にする, 47
- プリンタポート特性の調整 (概要), 133
- プリンタ名, 46
- フルクラッシュダンプディレクトリからの復元, 387
- プログラム
 - 強制終了, 394
 - ディスク依存度, 339
 - 割り込み, 394
- プログラムの強制終了, 394
- プログラムの割り込み, 394
- プロセス
 - init, 277
 - nice 番号, 314, 325, 327
 - nice番号, 325
 - proc ツールコマンド, 316
 - proc ツールコマンドを使用した情報の表示, 317
 - proc ツールを使用した情報の表示, 316
 - アカウントिंगユーティリティ, 277, 279, 293, 294
 - アドレス空間マップ, 317
 - アプリケーションスレッド, 309, 311
 - 一時停止, 317
 - 強制終了, 320
 - 現在の作業ディレクトリ, 317, 318
 - 構造体, 310, 314
 - 再開, 317
 - シグナルの動作, 317
 - 終了, 317, 320
 - 情報の表示, 293, 297, 298, 314, 321, 322
 - LWP, 317
 - ps コマンド, 314, 316
 - 実行中のプロセスの表示, 316
 - 日次利用状況レポート, 294
 - プロセスの表示, 315
 - 情報の表示 (方法), 318
 - スケジューリングクラス, 321, 322, 323, 324, 325
 - 情報の表示, 314
 - 優先順位の変更, 325
 - スタックトレース, 317
 - 制御, 319
 - 制御がきかない, 327
 - 追跡フラグ, 317
 - ツールコマンド, 317
 - ツリー, 317, 318

プロセス (続き)
定義, 309
開いたファイルの `fstat` 情報と `fcntl` 情報, 317
開いたファイルの `fstat` 情報と `fcntl` 情報, 317, 318
開いたファイルの `fstat` と `fcntl` 情報, 317
問題の解決, 327
優先順位, 321, 322, 323, 325
概要, 325
情報の表示, 314
タイムシェアリングプロセス優先順位の変更, 325
変更, 325
用語, 309, 311
リンクされたライブラリ, 317
プロセスアカウンティング, 277, 279, 293, 294
プロセスの強制終了, 320
プロセスの終了, 317
プロセスファイルシステム (PROCFS), 316
プロセス別コアファイルの設定, `coreadm` を使用した設定, 376

へ 変更

`crontab` ファイル, 262, 263
`dumpadm` を使用したクラッシュダンプ構成, 385
印刷要求の優先順位 (概要), 94
印刷要求の優先順位 (方法), 103
各ユーザー用のディスク割り当て制限, 253, 254
時間, 224
システムのホスト名, 225
スケジューリングクラス, 324, 325
その日のメッセージ, 225
日時, 224
フォーム用紙, 122
優先順位, 323, 325
タイムシェアリングプロセス, 325
弱い期間制限, 252, 253

編集

`crontab` ファイル, 262, 263

ほ

ポート, 47
状態 (表), 212
初期化プロセス, 197
初期化 (方法), 190
定義, 182
ポートモニター
`ttymon` と `listen` (定義), 198
`ttymon` と `listen` (定義), 183
状態 (表), 211
定義, 183
ボーレート設定, 408
ホスト常駐フォント, ダウンロード (概要), 129
ホスト名, 変更, 225

ま

毎週の作業 (`crontab` によるスケジューリング), 258
毎月の作業 (`crontab` によるスケジューリング), 258
毎日の作業 (`crontab` でスケジューリング), 258
待ち行列, 表示, 271

む

無効にする
`consadm` コマンドを使用した補助コンソール, 374
クラッシュダンプの保存, 388
特定のユーザーのディスク割り当て制限, 254, 255

め

メタデータ, 339
メモリ?
共有
プロセス仮想メモリ, 311
メモリー
仮想
プロセス, 311
情報の表示, 223, 294, 298
情報表示の例, 223

メモリー (続き)
プロセス構造体, 310

も

文字セット
選択可能, 106, 107
ソフトウェア, 106
ハードウェア, 106, 107, 108
番号, 106

モデム

管理用ツール, 184
異なる使用方法, 182
シリアルポートツールの概要, 186
シリアルポートツールのモデムテンプレート, 186
設定 (方法), 189
着信専用サービス, 182
定義, 182
発信専用サービス, 182
発着信両用サービス, 182, 198

問題の解決, 409

tty 回線, 293
印刷の問題, 405, 424
ソフトウェアパッケージのインストール/削除, 449
プロセス, 294, 327

ゆ

有効にする

consadm コマンドを使用した補助コンソール, 372
coreadm を使用したグローバルコアファイルパス, 379
coreadm を使用したプロセス別コアファイルパス, 379
クラッシュダンプを有効にする, 388
システムリブート後の補助コンソール, 373
ディスク割り当て制限, 249

ユーザー

印刷要求の取り消し (方法), 100
プリンタへのアクセス, 46
プリンタへのアクセス (概要)
lpadmin コマンドによる制限, 91

プリンタへのアクセス (概要) (続き)

Solaris プリンタマネージャによる制限, 91

ユーザーディスク割り当て制限

各ユーザー用に変更, 253
設定, 246, 247
超過の確認, 250
特定のユーザーについて無効にする, 254, 255

弱い期間制限, 252

ユーザープロセス

CPU 使用, 294
優先順位, 321
優先順位の変更, 325
ユーザーへの課金, 279, 283
ユーザーモード優先順位, 321
ユーザー料金, 279, 283, 294

ユーザーログイン

最終ログインの監視, 290, 297, 301
時間の監視, 277, 290, 294
ログインの数, 294

ユーザー割り当て制限, 個々のユーザーの変更, 254

優先順位 (プロセス)

概要, 321, 325
グローバル, 321, 322
最高位, 323
指定, 323
情報の表示, 314, 322
スケジューリングクラス, 323
変更, 323, 325
タイムシェアリングプロセス, 325
ユーザーモード優先順位, 321

よ

要求 ログ, 172

容量 (ディスク)

空き容量, 228, 337
最適化, 227

読み取りシステムコール

統計情報, 296, 298

弱い期間制限

超過, 252
変更, 252, 253

り

- リアルタイムプロセス
 - クラスの変更, 324, 325
 - 優先順位, 323
- リスト
 - ファイルとディレクトリ, 230, 231, 236
- レポート
 - 監視, 277, 279, 290, 293
 - クラッシュ後の失敗, 389
- リモート印刷
 - 処理図, 175
 - ユーザー料金の計算, 283
- 料金 (ユーザー), 279, 283, 294
- 履歴ログ (印刷要求), 164

る

- ルート crontab ファイル, 278
- ルートパスワードの復元
 - IA, 392
 - SPARC, 391
- ルートパスワード、忘れた
 - IA, 392
 - SPARC, 391

ろ

- ローカル印刷, 172
- ローカルプリンタ
 - 設定用作業マップ, 54
 - 定義, 42
- ローカルプリンタまたは接続プリンタ, Solaris
 - プリンタマネージャによる追加 (方法), 59
- ログイン監視
 - 最終ログイン, 290, 301
- ログインの監視
 - 最終ログイン, 297
 - 使用時間, 277, 279, 294
 - ログインの数, 294
- ログファイル, 172
 - LP 印刷サービス, 163, 407
 - 印刷待ち行列, 164
 - 印刷要求履歴ログ, 164
 - 自動削除, 263
 - 消去, 172
 - 要求, 172

ろ

- ログファイル (続き)
 - 要求ログのコード, 165
- ロック ファイル, 285

わ

- 忘れたルートパスワード
 - IA, 392
 - SPARC, 391
- 割り当て制限
 - 概要, 241
 - 個々のユーザーの変更, 254
 - 無効, 244, 255
 - 有効, 244
 - 有効、例, 249
 - ユーザー
 - 個々のユーザーの変更, 254
- 割り当て制限を無効にする, 244, 255
- 割り当て制限を有効にする, 244
- 割り当て制限を有効にする、例, 249

