

HP OpenVMS システム管理者マニュアル (下巻)

HP 部品番号: AA-PZWNK-TE
2011 年 12 月
第 1.2 版



© Copyright 2011 Hewlett-Packard Development Company, L.P.

著作権情報

本書の著作権は Hewlett-Packard Development Company, L.P. が保有しており、本書中の解説および図、表は Hewlett-Packard Development Company, L.P. の文書による許可なしに、その全体または一部を、いかなる場合にも再版あるいは複製することを禁じます。

日本ヒューレット・パカードは、弊社または弊社の指定する会社から納入された機器以外の機器で対象ソフトウェアを使用した場合、その性能あるいは信頼性について一切責任を負いかねます。

本書に記載されている事項は、予告なく変更されることがありますので、あらかじめご承知おきください。万一、本書の記述に誤りがあった場合でも、弊社は一切その責任を負いかねます。

本書で解説するソフトウェア (対象ソフトウェア) は、所定のライセンス契約が締結された場合に限り、その使用あるいは複製が許可されません。

Microsoft, Windows, Microsoft NT, および Microsoft XP は、Microsoft Corporation の米国における登録商標です。Microsoft Vista は、Microsoft Corporation の米国ならびに他の国における登録商標または商標です。

Intel, Pentium, Intel Inside は米国 Intel 社の登録商標です。UNIX, The Open Group は、The Open Group の米国ならびに他の国における商標です。Kerberos は、Massachusetts Institute of Technology の商標です。

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

UNIX is a registered trademark of The Open Group. Java is a US trademark of Sun Microsystems, Inc. Microsoft, Windows, and Windows NT are U.S. registered trademarks of Microsoft Corporation.

原典

『HP OpenVMS System Manager's Manual, Volume 2: Tuning, Monitoring, and Complex Systems』 © 2005 Hewlett-Packard Development Company, L.P.

目次

まえがき.....	19
対象読者.....	19
本書の構成.....	19
関連資料.....	19
本書で使用する表記法.....	20
1 システム・パラメータの管理.....	23
1.1 システム・パラメータについて.....	24
1.1.1 省略時の値, 現在値, アクティブ値.....	25
1.1.2 ページとページレット.....	26
1.2 パラメータ値の標準的な変更方法.....	26
1.3 AUTOGEN で使用するためのカスタマイズ済みパラメータ設定の変換.....	27
1.4 AUTOGEN コマンド・プロシージャについて.....	30
1.4.1 AUTOGEN フィードバックについて.....	32
1.4.2 フィードバック・レポート (AGENS\$PARAMS.REPORT) について.....	33
1.4.3 AUTOGEN のフェーズについて.....	37
1.4.4 AUTOGEN パラメータ・ファイル (MODPARAMS.DAT) について.....	38
1.5 AUTOGEN によるシステム・パラメータの変更.....	39
1.5.1 MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御.....	40
1.5.1.1 ADD_ 接頭辞を使って値を大きくする方法.....	41
1.5.1.2 MIN_ 接頭辞による最小値の指定方法.....	41
1.5.1.3 MAX_ 接頭辞による最大値の指定方法.....	41
1.5.1.4 絶対値の指定方法.....	42
1.5.1.5 VAXcluster ノード数の定義方法 (VAX のみ).....	42
1.5.1.6 イーサネット・アダプタの数を定義する方法 (VAX のみ).....	42
1.5.1.7 メモリを追加する前にあらかじめパラメータ値を設定する方法 (VAX のみ).....	42
1.5.1.8 DECnet に関連するパラメータを上書きする方法.....	43
1.5.1.9 NPAGEDYN と NPAGEVIR の値の設定.....	43
1.5.2 AUTOGEN フィードバックに必要な最小稼働時間の指定 (VAX のみ).....	43
1.5.3 外部パラメータ・ファイルの MODPARAMS.DAT への取り込み.....	43
1.5.4 DCL 文のログの停止.....	44
1.6 AUTOGEN レポートの自動化.....	44
1.6.1 AUTOGEN レポートを調べた後のパラメータ値の変更.....	46
1.7 SYSMAN ユーティリティによるシステム・パラメータの管理.....	47
1.7.1 パラメータ値と SYSMAN について.....	47
1.7.2 SYSMAN によるパラメータ値の表示.....	48
1.7.3 SYSMAN によるパラメータ・ファイルの変更.....	49
1.7.4 SYSMAN によるアクティブ値の変更.....	49
1.8 SYSGEN ユーティリティによるシステム・パラメータの管理.....	51
1.8.1 パラメータ値と SYSGEN について.....	51
1.8.2 SYSGEN によるパラメータ値の表示.....	52
1.8.3 SYSGEN によるシステム・パラメータ・ファイルの変更.....	54
1.8.4 SYSGEN によるアクティブ値の変更.....	54
1.8.5 SYSGEN による新規パラメータ・ファイルの作成.....	55
1.9 会話型ブートによるシステム・パラメータの変更.....	57
1.10 BAP システム・パラメータのチューニング.....	57
2 ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの管理.....	61
2.1 ダンプ・ファイルについて.....	62

2.1.1 ページ・ファイルを使ってシステム・クラッシュ・ダンプを格納する方法.....	63
2.1.2 システム・ダンプの種類.....	64
2.2 ページ・ファイルおよびスワップ・ファイルについて.....	65
2.3 ページ・ファイルとスワップ・ファイルに関する情報の表示.....	66
2.4 ダンプ・ファイル, ページ・ファイル, スワップ・ファイルのサイズの机上計算.....	67
2.4.1 システム・ダンプ・ファイルのサイズの計算.....	67
2.4.2 エラー・ログ・ダンプ・ファイルのサイズの計算.....	68
2.4.3 ページ・ファイルのサイズの計算.....	69
2.4.3.1 ページ・ファイル・サイズの表し方.....	70
2.4.3.2 ページ・ファイル使用状況の監視.....	70
2.4.3.3 ページ・ファイル空間の制限.....	71
2.4.4 スワップ・ファイル・サイズの計算.....	71
2.4.4.1 スワップ・ファイル・サイズの表し方.....	71
2.4.4.2 スワップ・ファイルの使用状況の監視.....	71
2.5 ディスク空間が十分でない場合のシステム・ダンプ・ファイル・サイズの最小化.....	72
2.5.1 選択型システム・ダンプでの情報の順序.....	73
2.5.2 選択型システム・ダンプにプロセスが書き込まれる順序の微調整 (Alpha および I64).....	74
2.6 システム・ディスクへのシステム・ダンプ・ファイルの書き込み.....	74
2.6.1 Alpha システムおよび I64 システムでのシステム・ディスクへのシステム・ダンプ.....	74
2.6.2 VAX システムでのシステム・ディスクへのシステム・ダンプ.....	76
2.7 代替ディスクへのシステム・ダンプ・ファイルの書き込み.....	76
2.7.1 Alpha システムおよび I64 システムでの DOSD の必要条件.....	76
2.7.2 VAX システムでの DOSD の必要条件.....	81
2.8 SDA によるクラッシュ・ダンプの内容の分析.....	82
2.9 SDA CLUE コマンドによるクラッシュ・ダンプ・ファイルの分析 (Alpha および I64).....	82
2.9.1 CLUE について (Alpha および I64).....	82
2.9.2 SDA CLUE コマンドによるデータの表示 (Alpha および I64).....	83
2.9.3 ダンプ・オフ・システム・ディスクと SDA CLUE の使用 (Alpha および I64).....	83
2.10 CLUE を使用して, クラッシュ・ダンプに関する履歴情報を得る方法 (VAX のみ).....	83
2.10.1 CLUE について (VAX のみ).....	84
2.10.2 CLUE を使用したデータの表示 (VAX のみ).....	84
2.11 システム障害後のシステム・ダンプ・ファイルの内容の保存.....	85
2.12 システム・ダンプ・ファイルをテープまたはディスクへコピーする.....	86
2.13 ページ・ファイルからのダンプ情報の解放.....	86
2.13.1 VAX システム, Alpha システム, および I64 システムでのダンプ情報の解放.....	86
2.13.2 VAX システム, Alpha システム, および I64 システムでダンプ情報を解放するための 使用上の注意.....	88
2.14 ページ・ファイルとスワップ・ファイルのインストール.....	88
2.14.1 会話型のインストール.....	88
2.14.2 SYPAGSWPFILES.COM でのインストール.....	89
2.15 ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの削除.....	90
2.16 ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの作成と変更.....	90
2.16.1 AUTOGEN を使用 (標準的な方法).....	91
2.16.1.1 システム・ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの格納場 所の制御.....	92
2.16.1.2 MODPARAMS.DAT でのページ・ファイル, スワップ・ファイル, ダンプ・ファ イルのサイズの制御.....	92
2.16.2 SWAPFILES.COM の使用方法.....	94
2.16.3 SYSGEN の使用法.....	96
2.17 プロセス・ダンプについて.....	98
2.17.1 特権ユーザおよびプロセス・ダンプへのアクセスについて (Alpha および I64).....	98
2.17.2 プロセス・ダンプへのアクセスの許可 (Alpha および I64).....	99
2.17.3 プロセス・ダンプへのアクセスの制限 (Alpha および I64).....	99

3 性能の管理	101
3.1 性能管理について.....	101
3.2 作業負荷の把握.....	102
3.3 作業負荷管理方針の決定.....	103
3.4 作業負荷の配分.....	103
3.5 システム・チューニングについて.....	104
3.6 チューニングが必要な時期の予測.....	105
3.7 チューニング結果の評価.....	105
3.8 性能オプションの選択.....	105
3.9 システム・ライブラリの展開.....	108
3.9.1 ライブラリを展開するために使用可能なディスク領域の決定.....	108
3.9.2 ライブラリ復元ユーティリティ (LIBDECOMP.COM) の使用.....	108
3.9.2.1 LIBDECOMP.COM が動作するライブラリ.....	109
3.9.2.2 LIBDECOMP.COM の会話型での使用.....	110
3.9.2.2.1 ライブラリのリスト.....	111
3.9.2.2.2 ライブラリの展開 (復元).....	113
3.9.2.2.3 ライブラリの縮小 (圧縮).....	114
3.9.2.3 バッチ・モードでの LIBDECOMP.COM の使用.....	115
3.9.3 LIBRARY コマンドでの /DATA 修飾子の使用.....	115
3.10 INSTALL による既知イメージのインストール.....	116
3.10.1 イメージおよび既知イメージについて.....	116
3.10.2 既知ファイルのエントリについて.....	117
3.10.3 既知イメージに割り当てることができる属性.....	117
3.10.4 インストールするイメージの決定.....	118
3.10.5 イメージ起動性能を向上するためのイメージのインストール方法.....	119
3.10.6 共用アドレス・データを付けたイメージのインストール.....	119
3.10.6.1 システムが提供するイメージ.....	120
3.10.6.2 アプリケーション・イメージ.....	120
3.10.7 物理メモリを節約するイメージのインストール方法.....	120
3.10.8 イメージの特権を拡張するためのイメージのインストール方法.....	120
3.10.8.1 特権付き実行可能イメージ.....	121
3.10.8.2 特権付き共用可能イメージ.....	121
3.10.9 特権コンテキストでのイメージの起動方法.....	122
3.10.10 INSTALL によるファイル名の指定.....	122
3.10.11 INSTALL によるイメージのインストール.....	122
3.10.12 INSTALL による既知イメージの表示.....	123
3.10.13 共用可能イメージ・ファイルの論理名の定義.....	124
3.10.14 既知イメージの削除.....	124
3.11 予約メモリ・レジストリ.....	125
3.11.1 予約メモリ・レジストリの使用.....	125
3.11.1.1 予約メモリ・レジストリ・データ・ファイル.....	125
3.11.1.2 AUTOGEN.....	126
3.11.1.3 予約メモリ・レジストリへのエントリの追加.....	126
3.11.2 予約メモリ・レジストリからのエントリの削除.....	127
3.11.2.1 予約メモリの割り当て.....	127
3.11.2.2 予約メモリの解放.....	127
3.11.2.3 予約メモリの表示.....	128
3.11.2.4 予約メモリの使用.....	129
3.11.2.5 予約メモリの復帰.....	129
3.11.3 アプリケーションの構成.....	129
4 ファイル・システムのデータ・キャッシュの管理	131
4.1 キャッシングについて.....	131
4.2 ファイル・システムのデータ・キャッシュについて.....	131

4.3 クラスタ全体でのキャッシングの禁止.....	132
4.4 キャッシングを禁止した状態でのボリュームのマウント.....	133
4.5 XFC の管理 (Alpha および I64).....	133
4.5.1 XFC と古いバージョンとの相互動作の保証.....	134
4.5.2 キャッシュのサイズの管理.....	134
4.5.2.1 最小キャッシュ・サイズの制御.....	134
4.5.2.2 最大キャッシュ・サイズの制御.....	135
4.5.2.3 静的キャッシュ・サイズの許可.....	136
4.5.3 最大キャッシュ I/O サイズの管理.....	137
4.5.4 ファイルのキャッシングの禁止.....	137
4.5.5 先読みキャッシングの禁止.....	138
4.5.6 性能の監視.....	138
4.5.6.1 システム全体の統計情報.....	138
4.5.6.2 複合アーキテクチャ OpenVMS Cluster での XFC の使用.....	140
4.6 Virtual I/O Cache の管理.....	140
4.6.1 Virtual I/O Cache のしくみについて.....	141
4.6.2 Alpha システムまたは I64 システムでの VIOC の選択.....	141
4.6.3 キャッシュのサイズの管理.....	141
4.6.4 VIOC 統計情報の表示.....	142
4.6.5 VIOC の許可.....	143
4.6.6 VIOC が許可されているかどうかの確認.....	143
4.6.7 メモリの割り当てと VIOC.....	144
4.6.8 VIOC サイズの調整.....	144
4.6.9 VIOC および OpenVMS Cluster の構成.....	144
5 UETP によるシステムのテスト.....	145
5.1 概要.....	145
5.1.1 UETP について.....	145
5.1.2 UETP の使用方法 (概要).....	146
5.2 UETP を使用するための準備.....	147
5.2.1 ログイン.....	147
5.2.2 SYSTEST ディレクトリの使用方法.....	148
5.3 テストを行うデバイスの設定.....	148
5.3.1 デバイスのチェック.....	148
5.3.2 必要なシステム・ディスク領域.....	149
5.3.3 UETP のディスク上での動作.....	149
5.3.4 ディスク・ドライブの準備.....	149
5.3.5 磁気テープ・ドライブ.....	150
5.3.6 テープ・カートリッジ・ドライブ.....	150
5.3.7 コンパクト・ディスク・ドライブ.....	151
5.3.8 光ディスク・ドライブ.....	151
5.3.9 ターミナルおよびライン・プリンタ.....	151
5.3.10 DR11-W データ・インタフェース (VAX のみ).....	151
5.3.11 DRV11-WA データ・インタフェース (VAX のみ).....	152
5.3.12 DR750 または DR780 (DR32 インタフェース) (VAX のみ).....	152
5.3.13 2 台目の LPA11-K デバイス.....	153
5.3.14 テストを行わないデバイス.....	153
5.3.15 OpenVMS Cluster のテスト.....	153
5.3.16 小規模ディスク・システムのテスト方法.....	154
5.3.17 DECnet for OpenVMS フェーズ.....	154
5.3.18 DECnet Phase 5 の論理名.....	155
5.3.19 ベクタ・プロセッサおよび VVIEF (VAX のみ).....	156
5.4 バッチでの UETP の実行.....	156
5.5 UETP の起動.....	157

5.5.1 フェーズのサブセットの実行方法.....	157
5.5.2 1つのフェーズの実行と複数のフェーズの実行.....	157
5.5.3 ロード・テスト用のユーザ負荷の定義.....	158
5.5.4 レポート形式.....	158
5.5.4.1 長いレポート形式.....	158
5.5.4.2 短いレポート形式.....	158
5.6 UETP の動作の停止.....	159
5.6.1 Ctrl/Y の使用方法.....	159
5.6.2 DCL コマンドの使用方法.....	159
5.6.3 Ctrl/C の使用方法.....	159
5.7 トラブルシューティング：概要.....	160
5.7.1 エラーの記録と診断.....	160
5.7.2 UETP 出力の中断.....	160
5.7.3 画面に情報を表示する方法.....	161
5.7.4 画面表示の例 (VAX のみ).....	161
5.7.5 画面表示の例 (Alpha および I64).....	162
5.7.6 UETP イーサネット・テスト用の遠隔ノードの定義.....	163
5.7.7 ログ・ファイル.....	164
5.8 トラブルシューティング：考えられる UETP エラー.....	164
5.8.1 一般的な障害の概要.....	164
5.8.2 クォータ、特権、アカウントの間違い.....	165
5.8.3 UETINIT01 障害.....	166
5.8.4 UETVECTOR 障害 (VAX のみ).....	167
5.8.5 ディスク領域の不足.....	168
5.8.6 OpenVMS Cluster システムの設定の間違い.....	168
5.8.7 ロード・テスト中の問題.....	169
5.8.8 DECnet for OpenVMS エラー.....	170
5.8.9 記録されるが表示されないエラー.....	171
5.8.10 PCB またはスワップ・スロットの欠如.....	171
5.8.11 キーボードの応答がない、またはシステム・ディスクが動作しない.....	172
5.8.12 FAL オブジェクトに対する省略時のアクセス権の欠如.....	172
5.8.13 バグ・チェックおよびマシン・チェック.....	172
5.9 UETP テストおよびフェーズ.....	173
5.9.1 初期化フェーズ.....	173
5.9.2 デバイス・テスト・フェーズ.....	173
5.9.2.1 デバイス・フェーズの動作.....	173
5.9.2.2 単一デバイス・テストの実行.....	174
5.9.2.3 UETINIDEV.DAT の形式.....	174
5.9.2.4 ループ・モードによるテストの実行.....	175
5.9.2.5 個々のデバイス・テストの機能.....	175
5.9.3 システム・ロード・テスト・フェーズ.....	176
5.9.4 DECnet for OpenVMS テスト・フェーズ.....	177
5.9.4.1 環境.....	177
5.9.4.2 DECnet フェーズの動作.....	178
5.9.5 クラスタ統合テスト・フェーズ.....	179
6 システムに関する情報の入手.....	181
6.1 システム・ログ・ファイルについて.....	181
6.2 エラー・ログ機構.....	182
6.3 エラー・ログ・フォーマットの用法.....	183
6.3.1 ERRFMT プロセスの再起動.....	183
6.3.2 エラー・ログ・ファイルの管理.....	183
6.3.3 ERRFMT によるメールの送信.....	183
6.3.3.1 メールを送信するための ERRFMT の停止と再起動.....	184

6.3.3.2	メールを他のユーザに送信する方法.....	184
6.4	Error Log Viewer (ELV) の使用方法.....	184
6.4.1	Error Log Viewer (ELV) について.....	185
6.4.2	ELV の起動.....	185
6.4.3	主な ELV コマンド.....	186
6.4.4	TRANSLATE コマンドを使用した標準レポート.....	186
6.4.4.1	標準レポートの例.....	186
6.5	オペレータ・ログ・ファイルの設定、管理、プリント.....	187
6.5.1	オペレータ・ログ・ファイルについて.....	187
6.5.2	OPCOM メッセージについて.....	188
6.5.2.1	初期化メッセージ.....	188
6.5.2.2	デバイス状態メッセージ.....	188
6.5.2.3	ターミナルの使用可能または不能メッセージ.....	189
6.5.2.4	ユーザ要求とオペレータ応答メッセージ.....	190
6.5.2.5	ボリュームがマウントまたはディスマウントされたことを示すメッセージ.....	191
6.5.2.6	システム・パラメータ・メッセージ.....	191
6.5.2.7	機密保護アラーム・メッセージ.....	192
6.5.2.8	オペレータ・ログ・ファイルの内容.....	192
6.5.3	オペレータ・ログ・ファイルの設定.....	193
6.5.3.1	オペレータ・ログ・ファイルの新バージョンの作成.....	194
6.5.3.2	論理名の指定.....	195
6.5.4	オペレータ・ログ・ファイルの管理.....	196
6.5.5	オペレータ・ログ・ファイルのプリント.....	196
6.6	機密保護監査機構の使用法.....	197
6.6.1	機密保護監査機構について.....	197
6.6.1.1	機密保護監査ログ・ファイル.....	198
6.6.1.2	混合バージョン・クラスタの監査ログ・ファイル.....	198
6.6.2	機密保護監査情報の表示.....	199
6.6.3	監査の開始を遅らせる方法.....	199
6.6.4	その他のクラスに対して機密保護監査機構を使用する方法.....	199
6.6.5	機密保護監査機構の使用禁止.....	200
6.6.6	ターミナルを使用可能にして、アラーム・メッセージを受信する方法.....	200
6.6.7	機密保護レポートの作成.....	201
6.6.8	機密保護監査ログ・ファイルの新しいバージョンの作成.....	201
6.6.8.1	ログ・ファイルの新しいクラスタ全体としてのバージョンの作成.....	202
6.6.8.2	ログ・ファイルの新しいノード固有のバージョンの作成.....	202
6.7	オペレーティング・システムの性能の監視.....	202
6.7.1	MONITOR について.....	203
6.7.1.1	MONITOR クラス.....	203
6.7.1.2	データの表示.....	204
6.7.1.3	出力のタイプ.....	204
6.7.2	MONITOR の起動.....	205
6.7.3	システムの動作の表示.....	206
6.7.4	システムの動作の記録.....	207
6.7.5	システムの動作の表示と記録.....	208
6.7.6	記録した動作のプレイバック.....	208
6.7.7	記録した動作の遠隔プレイバック.....	210
6.7.8	記録ファイルの更新.....	210
6.7.9	MONITOR の継続実行.....	211
6.7.9.1	MONITOR.COM プロシージャの使用法.....	211
6.7.9.2	SUBMON.COM プロシージャの使用法.....	213
6.7.9.3	MONSUM.COM プロシージャの使用法.....	213
6.7.10	遠隔監視.....	215

7 資源使用状況の調査	217
7.1 会計情報ファイルについて.....	217
7.2 調査対象の資源の決定.....	218
7.3 調査対象の資源の制御.....	219
7.4 会計情報ファイルの更新.....	219
7.5 会計情報ファイルの移動.....	219
7.6 資源の使用量を示すレポートの作成.....	220
7.7 会計グループの設定.....	221
7.8 ディスク空間使用量の調査.....	222
8 OpenVMS Cluster の管理	223
8.1 OpenVMS Cluster システムについて.....	223
8.1.1 OpenVMS Cluster 環境の設定.....	224
8.1.2 クラスタ全体のシステム管理.....	225
8.2 HP OpenVMS Availability Manager によるデータの解析.....	226
8.3 SHOW CLUSTER ユーティリティの使用法.....	226
8.3.1 SHOW CLUSTER ユーティリティについて.....	226
8.3.2 SHOW CLUSTER コマンドの起動.....	228
8.3.2.1 画面に表示されていない情報の表示.....	228
8.3.2.2 連続表示セッションの終了.....	229
8.3.2.3 SHOW CLUSTER での修飾子の使用.....	230
8.3.3 レポートへの情報の追加.....	230
8.3.4 表示データの制御.....	231
8.3.4.1 データ表示中のコマンド入力.....	231
8.3.4.2 ブロードキャスト・メッセージの削除.....	232
8.3.4.3 画面のリフレッシュ.....	232
8.3.5 表示データの書式化.....	232
8.3.5.1 レポートからの情報の削除.....	232
8.3.5.2 フィールドと画面サイズの変更.....	233
8.3.5.3 レポートの移動.....	233
8.3.5.4 レポートのスクロール.....	235
8.3.6 SHOW CLUSTER スタートアップ初期化ファイルの作成.....	235
8.3.7 SHOW CLUSTER ユーティリティのコマンドを含むコマンド・プロシージャの使用.....	236
8.4 SYSMAN ユーティリティと OpenVMS Cluster 管理について.....	237
8.5 システム管理ユーティリティ (SYSMAN) による機密保護の管理.....	237
8.5.1 グループ番号およびパスワードの変更.....	238
8.6 SYSMAN DO コマンドによる OpenVMS Cluster の管理.....	238
9 ネットワークについて	243
9.1 OpenVMS ネットワーク・ソフトウェア・オプション.....	244
9.2 HP ネットワーク・ソフトウェアの選択.....	244
9.3 HP TCP/IP Services for OpenVMS について.....	246
9.3.1 OpenVMS Cluster システムのサポート.....	247
9.3.2 TCP/IP Services 管理ツールおよびユーティリティ.....	247
9.4 TCP/IP ネットワークに参加するための準備.....	247
9.5 TCP/IP Services のインストールおよび構成.....	248
9.6 TCP/IP Services の開始と停止.....	248
9.7 TCP/IP Services のドキュメント.....	248
9.8 DECnet-Plus for OpenVMS ネットワーク・ソフトウェア.....	249
9.8.1 DECnet-Plus のノード名.....	250
9.8.2 OpenVMS Cluster システムのサポート.....	251
9.8.3 DECnet-Plus 管理ツールおよびユーティリティ.....	251
9.9 DECnet-Plus ネットワークに参加するための準備.....	252

9.10 DECnet-Plus のインストールおよび構成.....	252
9.11 DECnet over TCP/IP の使用方法.....	252
9.12 DECnet Phase IV から DECnet-Plus への移行.....	253
9.13 DECnet-Plus の開始と停止.....	254
9.14 DECnet-Plus Documentation.....	254
10 ローカル・エリア・ネットワーク (LAN) ソフトウェアの管理.....	255
10.1 ローカル・エリア・ネットワーク.....	255
10.1.1 LAN の特性.....	256
10.1.1.1 イーサネット LAN.....	257
10.1.1.2 FDDI LAN.....	257
10.1.1.3 トークン・リング LAN.....	257
10.1.1.4 ATM LAN.....	257
10.1.2 LAN アドレス.....	257
10.2 ローカル・エリア・ネットワークの管理.....	258
10.3 LANACP LAN サーバ・プロセス.....	259
10.3.1 LANACP LAN サーバ・プロセスの実行.....	260
10.3.2 LANACP LAN サーバ・プロセスの終了.....	260
10.4 LANCP ユーティリティ.....	260
10.4.1 LANCP の起動と実行.....	260
10.4.2 LANCP コマンド.....	261
10.4.3 LANCP のその他の機能.....	262
10.5 LAN デバイスの管理.....	262
10.5.1 システム・デバイスの表示.....	263
10.5.2 デバイス特性の表示.....	263
10.5.3 デバイス特性の設定.....	265
10.5.3.1 イーサネット・デバイスでの、DEFINE DEVICE コマンドおよび SET DEVICE コマンドの修飾子.....	266
10.5.3.2 LAN フェイルオーバ・デバイスでの、DEFINE DEVICE コマンドおよび SET DEVICE コマンドの修飾子.....	267
10.5.3.3 FDDI デバイスでの、DEFINE DEVICE コマンドおよび SET DEVICE コマンドの修飾子.....	267
10.5.3.4 トークン・リング・デバイスでの、DEFINE DEVICE コマンドおよび SET DEVICE コマンドの修飾子.....	268
10.5.3.5 修飾子 (ATM デバイス).....	270
10.6 LAN デバイス・データベースの管理.....	272
10.6.1 LAN デバイス・データベース内のデバイスの表示.....	272
10.6.2 LAN デバイス・データベースへのデバイスの入力.....	273
10.6.3 LAN デバイス・データベースからのデバイス・データの削除.....	273
10.7 LAN ノード・データベースの管理.....	274
10.7.1 LAN ノード・データベース内のノードの表示.....	274
10.7.2 LAN ノード・データベースへのノードの入力.....	275
10.7.3 LAN ノード・データベースからのノードの削除.....	276
10.8 LAN MOP.....	276
10.8.1 DECnet MOP との共存.....	276
10.8.2 DECnet MOP から LAN MOP への移行.....	277
10.8.3 CLUSTER_CONFIG_LAN.COM と LAN MOP.....	277
10.8.4 サテライト・ロードのサンプル.....	278
10.8.5 クロス・アーキテクチャ・ブート.....	279
10.9 LAN MOP ダウンライン・ロード・サービスの管理.....	279
10.9.1 MOP ダウンライン・ロード・サービスの許可.....	279
10.9.2 MOP ダウンライン・ロード・サービスの禁止.....	279
10.9.3 状態データとカウンタ・データの表示.....	279
10.9.4 個々のノードの状態とカウンタ・データの表示.....	280

10.9.5	カウンタ・データのクリア	281
10.9.6	OPCOM メッセージ	281
10.9.7	ロード・トレース機能	281
10.9.8	MOP コンソール・キャリア	282
10.9.9	MOP トリガ・ブート	282
10.10	LAN フェイルオーバについて	283
10.10.1	LAN フェイルオーバ・セットの作成	285
10.10.2	LAN フェイルオーバ・セットへの LAN デバイスの追加	285
10.10.3	LAN フェイルオーバ・セットからの LAN デバイスの削除	285
10.10.4	LAN フェイルオーバ・セットの有効化	286
10.10.5	LAN フェイルオーバ・セットの無効化	286
10.10.6	LAN フェイルオーバ・デバイスの優先順位の設定	286
10.10.7	LAN フェイルオーバ・セットの packetsize の設定	286
10.10.8	LAN フェイルオーバの特性の表示	287
10.10.9	LAN フェイルオーバ・カウンタの表示	287
10.10.10	LAN フェイルオーバ・セットのチェック	287
10.10.11	LAN フェイルオーバの例	287
11	InfoServer システムの管理	289
11.1	InfoServer の機能について	289
11.1.1	複数のサーバの自動サービスについて	292
11.1.2	サービスの中断を少なくする高可用性機能	292
11.1.3	X ターミナル・クライアントのサポート	292
11.2	LASTport プロトコルについて	293
11.2.1	LASTport トランスポート・プロトコル	293
11.2.2	LASTport/Disk プロトコル	293
11.2.3	LASTport/Tape プロトコル	294
11.3	サーバ管理セッションの開始	294
11.3.1	サーバ管理コマンド	295
11.4	InfoServer Client for OpenVMS 機能について	296
11.5	LASTCP ユーティリティの機能について	297
11.5.1	LASTCP ユーティリティの起動と終了	297
11.5.2	LASTCP コマンドの要約	297
11.5.3	InfoServer Client for OpenVMS の自動起動	298
11.5.4	DECnet の起動または停止による InfoServer クライアントの起動失敗	299
11.5.5	構成済みであるが媒体に接続されていない複数コントローラ (Alpha および I64)	300
11.5.6	スタートアップの制限事項: PATHWORKS と RSM	301
11.5.7	スタートアップの制限事項: SYSMAN	301
11.5.8	ユーザ・アカウントの必要条件	301
11.5.9	システム・パラメータ MAXBUF の必要条件	301
11.6	LADCP ユーティリティの機能について	301
11.6.1	LADCP ユーティリティの起動と終了	302
11.6.2	LADCP コマンドの要約	302
11.6.3	InfoServer デバイスを自動的に使用可能にする方法	303
12	LAT ソフトウェアの管理	305
12.1	LAT プロトコルについて	305
12.1.1	LAT プロトコルの機能	306
12.1.2	LAT プログラムの利点	306
12.2	LAT ネットワークについて	307
12.2.1	サービス・ノード	307
12.2.1.1	サービスの種類	307
12.2.1.2	サービスの通知	308

12.2.1.3 プリント要求.....	308
12.2.2 ターミナル・サーバ・ノード.....	308
12.2.2.1 サービス・ノードの位置の特定.....	308
12.2.2.2 接続の設定.....	309
12.2.2.3 サービス提供側のノード.....	309
12.2.3 外部への接続が可能なノード.....	309
12.2.4 LAT ネットワークのコンポーネント.....	309
12.3 LAT 構成について.....	310
12.3.1 OpenVMS Cluster および DECnet と LAT との関係.....	310
12.3.1.1 同じコントローラ上で実行する LAT と DECnet.....	311
12.3.1.2 異なるコントローラ上で実行する LAT と DECnet.....	311
12.3.2 複数の LAN アダプタの使用.....	311
12.3.2.1 サポートされる構成.....	312
12.3.2.2 サポートされない構成.....	313
12.3.2.3 論理 LAT リンクの作成.....	313
12.3.2.4 パスの発見.....	314
12.3.2.5 LAT パラメータの変更.....	314
12.3.3 イーサネット/FDDI 構成での大容量のバッファ.....	315
12.4 LATCP ユーティリティについて.....	316
12.4.1 LATCP の起動と終了.....	317
12.4.2 LATCP コマンド.....	317
12.5 LAT プロトコルのスタートアップ.....	318
12.6 LAT 特性のカスタマイズ.....	320
12.6.1 付加サービスの定義.....	321
12.6.2 ポートの設定.....	321
12.6.2.1 プリンタの設定.....	322
12.6.2.2 特殊アプリケーション・サービスの設定.....	322
12.6.2.3 制限されたサービスの設定.....	322
12.6.3 外部からの接続要求のキュー登録.....	323
12.6.4 外部への LAT 接続の許可.....	324
12.6.5 LAT\$SYSTARTUP.COM プロシージャの変更例.....	324
12.7 LATACP データベースのサイズの管理.....	326

13 特殊処理環境の管理.....327

13.1 マルチプロセッシングについて.....	328
13.1.1 1 次プロセッサと 2 次プロセッサ.....	328
13.1.2 アベイラブル・セットとアクティブ・セット.....	328
13.1.3 プロセッサの機能.....	328
13.2 対称型マルチプロセッシング (SMP) 環境の管理.....	329
13.2.1 マルチプロセッシング環境の作成.....	329
13.2.2 マルチプロセッシング環境の監視.....	329
13.3 ベクタ処理について.....	330
13.3.1 VAX におけるベクタ処理のサポート (VAX のみ).....	330
13.3.2 VAX ベクタ命令エミュレーション機能 (VVIEF) (VAX のみ).....	331
13.4 ベクタ処理環境の管理 (VAX のみ).....	331
13.4.1 ベクタ処理サポート・コードのロード (VAX のみ).....	331
13.4.2 ベクタ処理システムの構成 (VAX のみ).....	331
13.4.3 ベクタ・プロセスの管理 (VAX のみ).....	332
13.4.3.1 システム資源とプロセス・クォータの調整 (VAX のみ).....	333
13.4.3.2 プロセスへのスカラ資源およびベクタ資源の配分 (VAX のみ).....	333
13.4.4 ACL によるベクタ・プロセッサへのアクセスの制限 (VAX のみ).....	333
13.4.5 ベクタ処理システムに関する情報の入手 (VAX のみ).....	334
13.4.5.1 DCL レキシカル関数 F\$GETJPI および F\$GETSYI (VAX のみ).....	334
13.4.5.2 SHOW CPU/FULL コマンド (VAX のみ).....	335

13.4.5.3 SHOW PROCESS コマンドと LOGOUT/FULL コマンド (VAX のみ).....	335
13.4.6 VAX ベクタ命令エミュレーション機能 (VVIEF) のロード (VAX のみ).....	335
14 DECdtm サービスの管理.....	337
14.1 トランザクション・ログについて.....	338
14.2 トランザクション・ログの計画.....	339
14.2.1 トランザクション・ログのサイズの決定.....	339
14.2.2 トランザクション・ログの格納場所の決定.....	339
14.3 DECnet-Plus ネットワークの計画.....	340
14.3.1 DECnet-Plus ネームスペースの計画.....	340
14.3.2 DCEnet-Plus ネットワークでの SCSNODE 名の計画.....	340
14.3.2.1 SCSNODE 名に関する規則.....	340
14.3.2.2 トランザクション・グループについて.....	341
14.4 トランザクション・ログの作成.....	342
14.5 トランザクション性能の監視.....	344
14.6 トランザクション・ログのサイズが十分かどうかのチェック.....	346
14.7 トランザクション・ログのサイズの変更.....	347
14.8 トランザクション・ログの移動.....	349
14.9 ディスクのディスマウント.....	352
14.10 ノードの追加.....	354
14.11 ノードの削除.....	356
14.12 DECdtm サービスの停止.....	357
14.13 DECdtm サービスの開始.....	358
14.14 XA Gateway の使用 (Alpha および I64).....	359
14.14.1 ゲートウェイの設定.....	360
A Files-11 ディスク構造.....	363
A.1 ディスクの概念.....	363
A.1.1 ディスクの論理的構造.....	363
A.1.2 ディスクの物理的構造.....	364
A.2 Files-11 構造.....	364
A.2.1 ファイル識別子 (FID).....	365
A.2.2 ODS ディレクトリ階層.....	365
A.3 予約ファイル.....	365
A.3.1 索引ファイル INDEXF.SYS.....	366
A.3.1.1 ブート・ブロック.....	367
A.3.1.2 ホーム・ブロック.....	367
A.3.1.3 ファイル・ヘッダ.....	367
A.3.2 ストレージ・ビットマップ・ファイル BITMAP.SYS.....	368
A.3.3 不良ブロック・ファイル BADBLK.SYS.....	369
A.3.4 マスタ・ファイル・ディレクトリ.....	369
A.3.5 コア・イメージ・ファイル CORIMG.SYS.....	369
A.3.6 ボリューム・セット・リスト・ファイル VOLSET.SYS.....	369
A.3.7 継続ファイル CONTIN.SYS.....	369
A.3.8 バックアップ・ログ・ファイル BACKUP.SYS.....	369
A.3.9 保留不良ブロック・ログ・ファイル BADLOG.SYS.....	369
A.3.10 クォータ・ファイル QUOTA.SYS.....	369
A.3.11 ボリューム機密保護プロファイル SECURITY.SYS.....	370
A.4 Files-11 ODS レベル 1 と 2,5 の違い (VAX のみ).....	370

B 時差係数 (TDF) 表.....	371
C OpenVMS に用意されているタイムゾーン.....	375
用語集.....	385
索引.....	401

目次

1-1	パラメータの古い値と新しい値.....	29
1-2	SYSMAN パラメータの一時値, アクティブ値, 現在値.....	48
1-3	SYSGEN パラメータの一時値, アクティブ値, 現在値.....	52
8-1	省略時の SHOW CLUSTER レポート.....	227
8-2	CLUSTER が表示されている SHOW CLUSTER レポート.....	231
10-1	LAN フェイルオーバ.....	288
11-1	InfoServer システムによるクライアントへのサービスの提供.....	291
12-1	LAT ネットワークの構成例.....	310
12-2	複数アドレスの LAT 構成: 1 つの LAN にバージョンが混在する LAT ノードがある場合.....	312
12-3	複数アドレスの LAT 構成: 2 つの LAN にバージョンが混在する LAT ノードがある場合.....	312
12-4	複数アドレスの LAT 構成: 2 つの LAN にバージョン 5.3 の LAT ノードがある場合.....	313
12-5	サポートされない複数アドレスの LAT 構成.....	313
12-6	LAT FDDI リングおよび大容量のバッファ.....	315
14-1	DECdtm サービスの管理.....	338
14-2	トランザクション・グループ.....	342
A-1	ファイルのエクステント.....	363
A-2	トラックとシリンダ.....	364

表目次

1-1	AUTOGEN の制御.....	32
1-2	AUTOGEN のフェーズ.....	37
1-3	SYSMAN PARAMETERS コマンド.....	47
1-4	システム・パラメータとともに使用される SYSGEN コマンド.....	51
2-1	物理ダンプと選択型ダンプの比較.....	64
2-2	ページングとスワッピングに関連する用語.....	65
2-3	物理システム・ダンプ・ファイルと選択型システム・ダンプ・ファイルの比較.....	72
2-4	ページ・ファイル, スワップ・ファイル, ダンプ・ファイル, エラー・ログ・ダンプ・ファイル空間の合計サイズを制御するシンボル.....	93
2-5	個々のページ・ファイルおよびスワップ・ファイルのサイズを制御するためのシンボル.....	94
3-1	LIBDECOMP.COM が認識できるライブラリ.....	109
3-2	既知イメージの属性.....	117
5-1	デバイス・テスト (VAX のみ).....	176
5-2	デバイス・テスト (Alpha のみ).....	176
6-1	システム・ログ・ファイル.....	181
6-2	エラー・ログ・サブシステムの構成要素.....	182
6-3	ELV が完全にサポートするイベント・タイプ.....	185
6-4	ELV が部分的にサポートするイベント・タイプ.....	185
6-5	主な ELV コマンド.....	186
6-6	標準レポートの詳細レベル.....	186
6-7	省略時のイベント・クラス.....	197
6-8	MONITOR クラスの種類.....	203
6-9	MONITOR クラス.....	203
6-10	MONITOR のためのコマンド・プロシージャ.....	211
6-11	OpenVMS Cluster システムでの遠隔の監視互換性.....	215
8-1	省略時の SHOW CLUSTER レポートに含まれるフィールド.....	227
8-2	SHOW CLUSTER で利用できる修飾子.....	230
8-3	SHOW CLUSTER レポートで入手可能なクラス情報.....	230
8-4	CONFIGURATION コマンド.....	238
9-1	OpenVMS システム用の HP ネットワーク・ソフトウェアの選択.....	245
9-2	HP ネットワーク・ソフトウェアの相互操作性オプション.....	246
9-3	HP TCP/IP Services for OpenVMS のドキュメント.....	249
9-4	DECnet-Plus for OpenVMS のドキュメント.....	254
10-1	LAN 媒体の特性.....	256
10-2	LAN システム管理の強化.....	259
10-3	LANACP システム論理名.....	260
10-4	LANCP ユーティリティの機能.....	260
10-5	LANCP ユーティリティの起動.....	261
10-6	LANCP コマンド.....	261
10-7	LAN デバイス.....	262
10-8	SHOW DEVICE コマンド修飾子.....	263
10-9	トークン・リング・デバイスの, 省略時の機能アドレス.....	269
10-10	/CLIP の構文.....	270
10-11	/CLIP のキーワード.....	271
10-12	/ELAN の構文.....	271
10-13	/ELAN のキーワード.....	271
10-14	LIST DEVICE および SHOW DEVICE コマンド修飾子.....	273
10-15	PURGE DEVICE 修飾子.....	274
10-16	DEFINE NODE および SET NODE コマンド修飾子.....	275
10-17	LIST NODE および SHOW NODE コマンド修飾子.....	280
10-18	CONNECT NODE コマンド修飾子.....	282
10-19	TRIGGER NODE コマンド修飾子.....	283

11-1	InfoServer コマンド.....	295
11-2	LASTCP コマンド.....	297
11-3	LADCP コマンド.....	303
12-1	LATCP コマンド.....	317
13-1	システム・パラメータ VECTOR_PROC の設定.....	331
A-1	予約ファイル.....	366
A-2	Files-11 索引ファイルの内容.....	366
A-3	ファイル・ヘッダのデータ・エリア.....	368
A-4	Files-11 構造レベル 1 のボリュームにおける制限.....	370
B-1	欧州の TDF.....	371
B-2	北米の TDF.....	371
B-3	中米および南米の TDF.....	372
B-4	アジアの TDF.....	372
B-5	南太平洋の TDF.....	372
B-6	南極の TDF.....	373

例目次

1-1	AUTOGEN フィードバック・レポートの例.....	34
1-2	AUTOGEN コマンド・プロシージャの例.....	45
6-1	標準レポートと要約レポート.....	187
6-2	オペレータ・ログ・ファイルの例 (SYS\$MANAGER:OPERATOR.LOG).....	193
6-3	MONITOR.COM プロシージャ.....	212
6-4	SUBMON.COM プロシージャ.....	213
6-5	MONSUM.COM プロシージャ.....	214

まえがき

『OpenVMS システム管理者マニュアル (下巻)』は、2 巻からなる『OpenVMS システム管理者マニュアル』の下巻です。

対象読者

本書は、OpenVMS のシステム管理者を対象に書かれています。

本書は、OpenVMS の概念および操作に精通している方を対象にしているため、OpenVMS に関する基本的なことからについては説明していません。

本書の構成

本書は、以下の章および付録から構成されています。

- 第1章 「システム・パラメータの管理」
- 第2章 「ページ・ファイル、スワップ・ファイル、ダンプ・ファイルの管理」
- 第3章 「性能の管理」
- 第4章 「ファイル・システムのデータ・キャッシュの管理」
- 第5章 「UETP によるシステムのテスト」
- 第6章 「システムに関する情報の入手」
- 第7章 「資源使用状況の調査」
- 第8章 「OpenVMS Cluster の管理」
- 第9章 「ネットワークについて」
- 第10章 「ローカル・エリア・ネットワーク (LAN) ソフトウェアの管理」
- 第11章 「InfoServer システムの管理」
- 第12章 「LAT ソフトウェアの管理」
- 第14章 「DECdtm サービスの管理」
- 第13章 「特殊処理環境の管理」
- 付録 A 「Files-11 ディスク構造」
- 付録 B 「時差係数 (TDF) 表」
- 付録 C 「OpenVMS に用意されているタイムゾーン」
- 「用語集」

関連資料

以下のマニュアルにも OpenVMS の情報が記載されており、『OpenVMS システム管理者マニュアル』と併用すると役立ちます。

- 『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』
- 『OpenVMS ユーザーズ・マニュアル』
- 『OpenVMS Software Overview』¹
- 『Upgrade and Installation Manual』 (使用中のシステムに対応する最新版)
- 『OpenVMS システム・セキュリティ・ガイド』
- 『OpenVMS Performance Management』
- 『OpenVMS Cluster システム』および『OpenVMS Cluster 構成ガイド』
- 『TCP/IP Services for OpenVMS インストール/コンフィギュレーション』
- 『TCP/IP Services for OpenVMS Management』
- 『TCP/IP Services for OpenVMS Management Command Reference』

1. このマニュアルはアーカイブ扱いです。

- 『TCP/IP Services for OpenVMS Tuning and Troubleshooting』
- 『DECnet-Plus for OpenVMS Introduction and User's Guide』
- 『DECnet-Plus Planning Guide』
- 『DECnet-Plus for OpenVMS Applications Installation and Advanced Configuration』
- 『DECnet-Plus Network Control Language Reference』

HP OpenVMS 製品およびサービスについての追加情報は、次の Web サイトを参照してください。

<http://www.hp.com/jp/openvms/> (日本語)

<http://www.hp.com/go/openvms> (英語)

本書で使用する表記法

本書では、次の表記法を使用しています。

表記法	意味
Ctrl/x	Ctrl/x という表記は、Ctrl キーを押しながら別のキーまたはポインティング・デバイス・ボタンを押すことを示します。
PF1 x	PF1 x という表記は、PF1 に定義されたキーを押してから、別のキー (x) またはポインティング・デバイス・ボタンを押すことを示します。
Return	例の中で、キー名が太字で書かれている場合には、そのキーを押すことを示します。
...	例の中の水平方向の反復記号は、次のいずれかを示します。 <ul style="list-style-type: none"> • 文中のオプションの引数が省略されている。 • 直前の 1 つまたは複数の項目を繰り返すことができる。 • パラメータや値などの情報をさらに入力できる。
.	垂直方向の反復記号は、コードの例やコマンド形式の中の項目が省略されていることを示します。このように項目が省略されるのは、その項目が説明している内容にとって重要ではないからです。
()	コマンドの形式の説明において、括弧は、複数のオプションを選択した場合に、選択したオプションを括弧で囲まなければならないことを示しています。
[]	コマンドの形式の説明において、大括弧で囲まれた要素は省略可能な選択肢です。項目をすべて選択しても、いずれか 1 つを選択しても、あるいは 1 つも選択しなくても構いません。コマンド行には、大括弧は入力しないでください。ただし、OpenVMS ファイル指定のディレクトリ名の構文や、割り当て文の部分文字列指定の構文の中では、大括弧も含めて入力しなければなりません。
	コマンド形式の説明では、縦棒は大括弧や中括弧内の選択肢を区切っています。大括弧内の選択肢は省略可能ですが、中括弧内の選択肢は少なくとも 1 つ選択する必要があります。コマンド行には、縦棒は入力しないでください。
{ }	コマンドの形式の説明において、中括弧で囲まれた選択肢は必須なので、いずれか 1 つを選択しなければなりません。コマンド行には、中括弧は入力しないでください。
太字体	太字体のテキストは、新しい用語、引数、属性、条件を示しています。
<i>italic type</i>	イタリック体は、重要な情報を示します。また、システム・メッセージ (たとえば内部エラー number)、コマンド・ライン (たとえば /PRODUCER=name)、コマンド・パラメータ (たとえば device-name) などの変数を示す場合にも使用されます。
UPPERCASE TYPE	英大文字は、コマンド、ルーチン名、ファイル名、システム特権の短縮形を示します。
Example	この字体は、コード例、コマンド例、および対話型の画面表示を示します。テキスト内では、この字体は URL、UNIX® のコマンドとパス名、PC ベースのコマンドとフォルダ、および C プログラミング言語の要素も示します。

表記法	意味
-	コマンド形式の記述の最後，コマンド・ライン，コード・ラインにおいて，ハイフンは，要求に対する引数とその後の行に続くことを示します。
数字	特に明記しない限り，本文中の数字はすべて 10 進数です。10 進数以外 (2 進数，8 進数，16 進数) は，その旨を明記してあります。

第1章 システム・パラメータの管理

システムをインストールまたはアップグレードすると、システム・パラメータの値は弊社の提供するコマンド・プロシージャ SYS\$UPDATE:AUTOGEN.COM (AUTOGEN) によって自動的に設定されます。定期的に AUTOGEN を使用し、ハードウェア構成とシステムの作業負荷に合わせて、システム・パラメータの値を調整してください。

この章の内容

この章では、次の作業について説明します。

作業	参照箇所
AUTOGEN で使用するためのカスタマイズ済みパラメータ設定の変換	1.3 項「AUTOGEN で使用するためのカスタマイズ済みパラメータ設定の変換」
AUTOGEN によるシステム・パラメータ値の変更 (標準的な方法)	1.5 項「AUTOGEN によるシステム・パラメータの変更」
MODPARAMS.DAT による AUTOGEN パラメータ設定の制御	1.5.1 項「MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御」
AUTOGEN レポートの自動化	1.6 項「AUTOGEN レポートの自動化」
SYSMAN によるシステム・パラメータの管理	1.7 項「SYSMAN ユーティリティによるシステム・パラメータの管理」
SYSGEN によるシステム・パラメータの管理	1.8 項「SYSGEN ユーティリティによるシステム・パラメータの管理」
会話型ブートによるシステム・パラメータの管理	1.9 項「会話型ブートによるシステム・パラメータの変更」

さらに、次の項目について説明します。

項目	参照箇所
システム・パラメータ	1.1 項「システム・パラメータについて」
システム・パラメータの省略時の値、現在値、アクティブ値	1.1.1 項「省略時の値、現在値、アクティブ値」
ページとページレット	1.1.2 項「ページとページレット」
システム・パラメータ値の標準的な変更方法	1.2 項「パラメータ値の標準的な変更方法」
AUTOGEN.COM コマンド・プロシージャ	1.4 項「AUTOGEN コマンド・プロシージャについて」
AUTOGEN フィードバック	1.4.1 項「AUTOGEN フィードバックについて」
AUTOGEN フィードバック・レポート (AGEN\$PARAMS.REPORT)	1.4.2 項「フィードバック・レポート (AGEN\$PARAMS.REPORT) について」

項目	参照箇所
AUTOGEN のフェーズ	1.4.3 項「AUTOGEN のフェーズについて」
AUTOGEN パラメータ・ファイル (MODPARAMS.DAT)	1.4.4 項「AUTOGEN パラメータ・ファイル (MODPARAMS.DAT) について」

1.1 システム・パラメータについて

システムがどのように機能するかは、**システム・パラメータ**の値によって制御されます。システム・パラメータは、広範囲のシステム機能を制御します。次に、システム・パラメータで制御できる機能の一部を紹介します。

- メモリ管理
- スケジューリング
- 機密保護属性
- システム・キャッシュ
- ウィンドウ・システム選択
- ターミナル構成
- VAXcluster または OpenVMS Cluster システム属性

『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』では、システム・パラメータの一覧を示し、各パラメータについて説明しています。

ディストリビューション・キットで提供されるシステム・パラメータの**省略時の値**は、どんな構成でもブートするように設定されています。システムをインストールまたはアップグレードすると、SYS\$UPDATE:AUTOGEN.COM コマンド・プロシージャが実行され、システム構成の評価と主な作業負荷の予想を行い、必要に応じてシステム・パラメータの値を調整します。各システム・パラメータには、許容値の範囲を定義する最大値と最小値が設定されています。

パラメータ・タイプ

各システム・パラメータのタイプは、次の 1 つまたは複数になります。

タイプ	説明
ダイナミック	ダイナミック・システム・パラメータの値は、メモリ内のアクティブ値を変更することによりシステムの稼働中に変更できる。これに対してダイナミック・パラメータ以外のパラメータの値を変更する場合、パラメータ・ファイルに格納されている現在値を変更した後、システムをリブートして変更した値を有効にしなければならない。アクティブ値と現在値についての詳細は、1.1.1 項「省略時の値、現在値、アクティブ値」を参照。
汎用	汎用パラメータの値は、ブート時におけるデータ構造の作成と初期化に影響する。
メジャー	メジャー・パラメータは変更する必要性が最も高い。
特殊	特殊パラメータは、弊社で使用するためだけに用意される。これらのパラメータは、弊社のエンジニアから指示があった場合、または弊社の製品のインストール・ガイドまたはリリース・ノートに示されている場合にのみ変更する。

機能別のパラメータ・カテゴリ

システム・パラメータは、その機能別に次のカテゴリに分けることができます。

カテゴリ	機能
ACP	ファイル・システム・キャッシュおよび Files-11 XQP (拡張 QIO プロシージャ)、すなわち補助制御プロセス (ACP) に関連するパラメータ。
クラスタ	VAXcluster または OpenVMS Cluster 操作を制御するパラメータ。
ジョブ	ジョブを制御するパラメータ。
LGI	ログイン・セキュリティを制御するパラメータ。
マルチプロセッシング	対称型マルチプロセッシングに関連するパラメータ。
PQL	プロセス作成上の制限とクォータに関連するパラメータ。
RMS	OpenVMS レコード管理サービス (RMS) に関連するパラメータ。
SCS	システム通信サービス (SCS) とポート・ドライバの操作を制御するパラメータ。SCS 操作に影響するパラメータの先頭には SCS という文字列が付けられる。
SYS	システム操作全体に影響するパラメータ。
TTY	ターミナルの動作に関連するパラメータ。
ユーザ定義	次のパラメータはユーザが定義できる。 <ul style="list-style-type: none">• USERID1 (ダイナミック・パラメータ)• USERID2 (ダイナミック・パラメータ)• USER3• USER4

1 ACP パラメータの多くは、Files-11 ディスク構造レベル 1 ディスクがマウントされている場合、あるいはマウント・コマンドで ACP が特に要求された場合にだけ適用できる。バージョン 4.0 より前のオペレーティング・システムでは、ファイルのオープンやクローズ、ウィンドウ切り替えといった、ファイル操作は補助制御プロセス (ACP) という別のプロセスが行う。バージョン 4.0 では、XQP (拡張 QIO プロシージャ) が導入され、これらの操作はシステム上の各プロセスが行うようになった。互換性のため、パラメータの名前は変更されていない。

1.1.1 省略時の値、現在値、アクティブ値

システムの各システム・パラメータは、次の 4 種類の値をとります。

値の種類	説明
省略時の値	システムにあらかじめ設定されている値で、サポートされるどの構成でもブートできるように設定されている。
現在値	ディスク上の省略時のパラメータ・ファイルに格納され、システムのブート時に使用される値。 VAX システムの省略時のパラメータ・ファイルは VAXVMSSYS.PAR。 Alpha システムの省略時のパラメータ・ファイルは ALPHAVMSSYS.PAR。 I64 システムの省略時のパラメータ・ファイルは IA64VMSSYS.PAR。
アクティブ値	メモリに格納され、システムの稼働中に使用される値。システムの稼働中に変更できるアクティブ値は、カテゴリがダイナミック・システム・パラメータであるシステム・パラメータの値に限られる。
他のパラメータ・ファイルに格納されている値	現在値を格納する省略時のパラメータ・ファイル以外にも、特別な目的のパラメータ・ファイルを作成できる。

ブート時、システムは現在値をメモリに読み込み、アクティブ値を作成します。変更がない限り、アクティブ値と現在値は同じです。

AUTOGEN コマンド・プロシージャの SETPARAM フェーズを実行すると、現在値が変更されます。

SYSMAN ユーティリティと SYSGEN ユーティリティにより、**現在値**と**アクティブ値**の両方を表示または変更できます。表示したり変更したい値を指定するには、USE コマンドおよび WRITE コマンドを使用します。

SYSMAN によるパラメータの管理についての詳細は、1.7 項「SYSMAN ユーティリティによるシステム・パラメータの管理」を参照してください。SYSGEN によるパラメータの管理についての詳細は、1.8 項「SYSGEN ユーティリティによるシステム・パラメータの管理」を参照してください。

1.1.2 ページとページレット

VAX システムでは、オペレーティング・システムは**ページ**と呼ばれる単位を使って、プロセスに対するメモリの割り当てや割り当て解除を行います。VAX システムのページ・サイズは 512 バイトです。システム・パラメータのいくつかはページ単位で割り当てられます。

Alpha システムおよび I64 システムでは、システム・パラメータのいくつかはページ単位で割り当てられますが、**ページレット**単位で割り当てられるシステム・パラメータもあります。

Alpha と I64 のどちらも、いろいろなページ・サイズをサポートしています。OpenVMS オペレーティング・システムでは、現在、Alpha システムと I64 システムで 8KB (8192 バイト) のページを使用しています。ページレットとは、512 バイトのメモリ単位です。Alpha システムまたは I64 システムの 1 ページレットは VAX の 1 ページと同じサイズです。また、ページ・サイズが 8KB の Alpha システムまたは I64 システムでは、16 ページレットが 1 ページと等しくなります。

パラメータ値、特にメモリ管理に関連するパラメータを調べる場合には、各パラメータに必要な割り当て単位に注意してください。1.7.2 項「SYSMAN によるパラメータ値の表示」と 1.8.2 項「SYSGEN によるパラメータ値の表示」では、パラメータ値とその割り当て単位を表示する方法が説明されています。

1.2 パラメータ値の標準的な変更方法

システム・パラメータの多くは、他のパラメータやシステムの性能に影響します。したがって、システム・パラメータを管理するときにはコマンド・プロシージャ

SYS\$UPDATE:AUTOGEN.COM (AUTOGEN) を使用してください。AUTOGEN については、1.4 項「AUTOGEN コマンド・プロシージャについて」を参照してください。

またシステム・パラメータの管理は、SYSMAN ユーティリティや SYSGEN ユーティリティを使って行うこともできます。パラメータ値の変更にこれらのユーティリティを使用するのは一般的ではありません。しかし、次の場合には使用できます。

- VAX, Alpha, または I64 システム上で、システム・パラメータとその値を表示する場合
- OpenVMS Cluster 環境のシステムのシステム・パラメータとその値を表示する場合
- 他のパラメータにほとんど影響のないパラメータを単独で、一時的に変更する場合



重要:

SYSMAN または SYSGEN を使ってパラメータ値を変更すると、AUTOGEN の実行時に、値が省略時の値に変更されたり再設定されることがあります。AUTOGEN を実行してもパラメータの変更内容が保持されるようにするためには、変更した値を AUTOGEN パラメータ・ファイル MODPARAMS.DAT に追加します。詳細は 1.5.1 項「MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御」を参照してください。

カスタマイズ済みのパラメータ設定を MODPARAMS.DAT に追加してなく、SYSMAN または SYSGEN を使用してパラメータを変更しようとする場合は、AUTOGEN を実行する前に 1.3 項「AUTOGEN で使用するためのカスタマイズ済みパラメータ設定の変換」の指示に従ってください。

1.3 AUTOGEN で使用するためのカスタマイズ済みパラメータ設定の変換

システムのチューニングには、できるだけ AUTOGEN コマンド・プロシージャを使用するようにしてください。システム管理ユーティリティ (SYSMAN) またはシステム生成ユーティリティ (SYSGEN) を使用してシステム・パラメータ値を変更し、その変更内容を AUTOGEN パラメータ・ファイル MODPARAMS.DAT に保存しなかった場合、これらの変更内容は次に AUTOGEN を実行したときに上書きされます。

以前に、SYSMAN または SYSGEN を使ってパラメータ値を変更した場合は、AUTOGEN で使用できるように、次の手順でパラメータ設定を変更します。この手順に従うと、AUTOGEN を実行してもその設定が維持されるように、カスタマイズ済みパラメータ設定を MODPARAMS.DAT に追加することができます。

この作業を行う場合は、その前に 1.4 項「AUTOGEN コマンド・プロシージャについて」を参照して、AUTOGEN、フィードバック、AUTOGEN パラメータ・ファイル MODPARAMS.DAT について理解しておいてください。

1. システムが現在使用しているパラメータ値を保存する。次に例を示す。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> PARAMETERS USE ACTIVE
SYSMAN> PARAMETERS WRITE SYS$SYSTEM:ノード名 _PARAMS_CURRENT.PAR
```

2. アクティブ・パラメータ値のリストをノード名 _PARAMS.OLD という ASCII ファイルに書き込む。次に例を示す。

```
SYSMAN> PARAMETERS SHOW/ALL/OUTPUT= ノード名 PARAMS.OLD
SYSMAN> PARAMETERS SHOW/SPECIAL/OUTPUT= ノード名 _PARAMS_SPECIAL.OLD
SYSMAN> EXIT
$ APPEND ノード名 _PARAMS_SPECIAL.OLD ノード名 _PARAMS.OLD
```

このファイルは、ステップ 6 で使用する。

3. AUTOGEN のパラメータ・ファイル SYS\$SYSTEM:MODPARAMS.DAT を編集してシンボルを定義し、以下の値を指定する。
 - AUTOGEN で算出されないパラメータ値 (SCSNODE, SCSSYSTEMID など)。AUTOGEN で算出されるパラメータについては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の AUTOGEN の説明の部分にある表を参照。
 - 使用するシステムの作業負荷に合うように調整する必要のある、すべてのパラメータ値 (GBLPAGES, GBLSECTIONS など)。

値の指定は、明示的に行うのではなく、MIN_ パラメータ名、MAX_ パラメータ名、ADD_ パラメータ名のいずれかの形式を使用してシンボルを定義する。次に例を示す。

```
$ EDIT SYS$SYSTEM:MODPARAMS.DAT
```

```
SCSNODE = "MYNODE"      ! Not calculated by AUTOGEN
SCSSYSTEMID = 10001     ! Not calculated by AUTOGEN
MIN_GBLPAGES = 10000    ! Needed for MCS, BLISS32, and ADA
MIN_GBLSECTIONS = 600  ! Needed for MCS, BLISS32, and ADA
```

MODPARAMS.DAT に加えた変更内容を分かりやすく示すために、各行にコメントを追加する。コメントの先頭には感嘆符 (!) を付ける。MODPARAMS.DAT にシンボルを定義する方法については、1.5.1 項「MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御」を参照。

4. AUTOGEN を実行する。ただし、リポートはしない。システムに応じて次のいずれかのコマンドを使用する。

- システムをブートしてから 24 時間以上通常の作業負荷を実行している場合は、次のコマンドを使用する。

```
$ @SYS$UPDATE:AUTOGEN SAVPARAMS SETPARAMS FEEDBACK
```

SAVPARAMS フェーズでは、稼働中のシステムに対する資源使用量についてのフィードバック情報を収集する。この情報は AUTOGEN が使用する。このコマンドにより作成される SYS\$SYSTEM:AGEN\$PARAMS.REPORT というフィードバック・レポートからピーク時の資源使用量を知ることができる。

- 以前に収集したフィードバック・ファイルを使いたい場合は、次のコマンドを使用する。

```
$ @SYS$UPDATE:AUTOGEN GETDATA SETPARAMS FEEDBACK
```

GETDATA フェーズから起動すると、AUTOGEN は現在のフィードバックを収集しない。

- 新しいシステムの場合 (すなわちフィードバックがない場合)、あるいはシステムが最後にブートして以来行ったアクティビティがほとんどないため (たとえば、週末にかかっていた場合など)、有効なフィードバック・ファイルがない場合には、次のコマンドを使用する。

```
$ @SYS$UPDATE:AUTOGEN GETDATA SETPARAMS CHECK_FEEDBACK
```

フィードバックが有効かどうかを AUTOGEN に判断させるには、CHECK_FEEDBACK を使用する。

5. 新しいパラメータ値のリストを ASCII ファイルに書き込む。次に例を示す。

```
SYSMAN> PARAMETERS USE CURRENT
SYSMAN> PARAMETERS SHOW /ALL /OUTPUT= ノード名 PARAMS.NEW
SYSMAN> PARAMETERS SHOW /SPECIAL /OUTPUT= ノード名 _PARAMS_SPECIAL.NEW
SYSMAN> EXIT
$ APPEND ノード名 _PARAMS_SPECIAL.NEW; ノード名 _PARAMS.NEW
```

6. 古いパラメータ値と新しいパラメータ値を比較する。次に例を示す。

```
$ DIFFERENCES/PARALLEL/OUTPUT= ノード名 _PARAMS.DIF/MATCH=5 -
_ $ ノード名 _PARAMS.OLD ノード名 _PARAMS.NEW
```

7. ステップ 6 で作成した差分ファイル (ファイル名の形式は ノード名 _PARAMS.DIF) を出力する。出力は読み易さを考えて 132 カラムのライン・プリンタで行う。
8. 各パラメータ名カラムの後の 2 つのカラムの数値を比較する。左側のカラムには古い値、右側のカラムには新しい値が示されている。図 1-1 「パラメータの古い値と新しい値」に、出力の例を示す。

図 1-1 パラメータの古い値と新しい値

パラメータ名	古い値				新しい値			
	77500	10000	512		81800	10000	512	
GBLPAGES	77500	10000	512		81800	10000	512	
SYSMWCNT	2400	500	40	1638	2800	500	40	1638
INTSTKPAGES	4	4	1		4	4	1	
BALSETCNT	250	16	4	819	250	16	4	819
WSMAX	32800	1024	60	20000	65500	1024	60	20000
NPAGEDYN	1944576	360000	16384		3000000	360000	16384	
NPAGEVIR	777328	1000000	16384		12000000	1000000	16384	
PAGEDYN	1516032	190000	10240		1780056	190000	10240	
VIRTUALPAGECNT	150000	9216	512	100000	270144	9216	512	100000
...			

9. MIN_, MAX_, ADD_ のいずれかが先頭に付いたシンボルを使って、MODPARAMS.DAT の調整を行う。たとえば、AUTOGEN により GBLPAGES に比較的小さい値が算出された場合、次のようにして、このパラメータに最小値を指定することができる。

```
MIN_GBLPAGES = 10000
```

ステップ 3 で MODPARAMS.DAT にパラメータ値を指定したのに、そのパラメータが変更されていない場合は、次の事柄をチェックする。

- パラメータ名のスペルが正しく完全である (省略されていない) こと。
MODPARAMS.DAT では、AUTOGEN はパラメータをシンボル割り当てとして認識する。スペルミスがあると、AUTOGEN はシンボルと対応するシステム・パラメータとを同一のものとして認識することができない。AUTOGEN により作成されるエラー・メッセージについては、AGEN\$FEEDBACK.REPORT を参照。
- 値が正しいこと。桁数を数え、コンマが入っていないことを確認する。
- このパラメータが MODPARAMS.DAT に一度だけしか指定されていないこと。
- パラメータがコメントアウトされていないこと。

ほとんどのパラメータは、新しい値の方が古い値よりも大きければ、AUTOGEN による設定を受け入れる。新しい値の方が古い値よりも小さい場合は、AUTOGEN の実行時にその資源を使用できなかったため、古い値を残しておくようにする。

たとえば、SYSMAN を使って、レイヤード製品に合うように GBLPAGES を 10,000 に増やしたのに、その変更内容を MODPARAMS.DAT に保存しなかった場合、AUTOGEN はシステムが必要とするのは 5,000 グローバル・ページだけであると算出する。この場合、AUTOGEN の実行後レポートすると、レイヤード製品の一部がインストールされずに、システム・メッセージ GPTFULL (global page table full) が表示されて、さらに GBLPAGES が必要であることが示される。

10. 満足できるパラメータ値が決まるまで、ステップ 3 以降を繰り返す。

必要ならば、MODPARAMS.DAT にさらに変更を加え、AUTOGEN を再度実行し、上述したようにその変更内容をチェックする。通常、AUTOGEN のチェックに 2 回パスすれば、このパラメータ値が確定し、リポートすることができる。

11. リポートする。リポートすると、新しいパラメータ値が有効になる。リポートには AUTOGEN は必要ないし、またすぐにリポートする必要もない。ただし、新しいパラメータ値が使用される前にリポートする必要がある。

システムがブートしない場合は、会話型ブートを行い、ステップ 1 で作成したバックアップ・パラメータ・ファイルを使用する。

```
SYSBOOT> USE SYS$SYSTEM:ノード名 _PARAMS_CURRENT.PAR
SYSBOOT> CONTINUE
```

CONTINUE コマンドを入力すると、AUTOGEN の実行前に保存したパラメータ値でシステムがブートされる。

システムのブート後、古いパラメータ値が必要になった場合は、次のコマンドを使用する。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> PARAMETERS USE SYS$SYSTEM:ノード名 _PARAMS_CURRENT.PAR
SYSMAN> PARAMETERS WRITE CURRENT
SYSMAN> EXIT
```

12. システムの資源が確実にシステムの作業負荷に一致するように、フィードバックを使って定期的に AUTOGEN を実行する。フィードバックを使った AUTOGEN の実行方法については、1.5 項「AUTOGEN によるシステム・パラメータの変更」を参照。

1.4 AUTOGEN コマンド・プロシージャについて

AUTOGEN コマンド・プロシージャ SYS\$UPDATE:AUTOGEN.COM は、ディストリビューション・キットで提供されます。AUTOGEN は、システムのインストールまたはアップグレード時に自動的に実行され、システム・パラメータを適切な値に設定します。また、システム・パラメータの値を再設定したり、ページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズを変更したりする場合にも、AUTOGEN を使用してください。新しい値は、システムのブート後、有効になります。

AUTOGEN は特定の重要なシステム・パラメータのみを算出します。AUTOGEN により算出されるシステム・パラメータの表が『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』の AUTOGEN の節にありますので参照してください。

どのような場合に AUTOGEN を実行するか？

AUTOGEN は次の場合に実行してください。

- インストールまたはアップグレード時。この場合は、インストレーション・プロシージャまたはアップグレード・プロシージャの一部として自動的に実行される。
- 作業負荷が大幅に変化した場合。
- オプションの (レイヤード) ソフトウェア製品を追加する場合。インストレーションの必要条件については、個々の製品のドキュメントを参照。レイヤード製品の中には、AUTOGEN によってパラメータ値やページ・ファイルおよびスワップ・ファイルのサイズを変更しなければならないものもある。AUTOGEN を使ってパラメータ値やページ・ファイルおよびスワップ・ファイルのサイズを調整する方法については、2.16.1 項「AUTOGEN を使用 (標準的な方法)」を参照。
- /SHARED 属性によりイメージをインストールする場合。この場合、追加されるグローバル・ページとグローバル・セクションに合わせてパラメータ GBLSECTIONS および GBLPAGES の値を大きくする必要がある。
- システムの作業負荷の変更を定期的に監視する場合。フィードバックを定期的にチェックし、システム・パラメータの変更を指示するように、AUTOGEN を自動化することができる。AUTOGEN をフィードバック・モードで定期的に実行し、フィードバック・レポートを該当する Mail アカウントに自動的に送信する、バッチ形式のコマンド・プロシージャについては、1.6 項「AUTOGEN レポートの自動化」を参照。
- 定期的に適切なスワップ・ファイル用のスペースを空ける。FEEDBACK オプションを使用して、システムが十分な時間 (24 時間以上) 立ち上がっていて、通常のロードであることを確認する。また、SYS\$SYSTEM:MODPARAMS.DAT ファイルが、決まった SWAPFILE 値を含んでいないことも確認する。値が決まっていると、AUTOGEN はスワップ・ファイルのサイズを正しく決定できない。

AUTOGEN の機能

AUTOGEN はフェーズ単位で実行されます。AUTOGEN をどのフェーズで実行させるかによって、次の動作の一部またはすべてが行われます。

- 次のデータの収集
 - 稼働中のシステムについてのフィードバック
 - システムのハードウェア構成

- ユーザが設定したパラメータの要件 (MODPARAMS.DAT より)
- 弊社が設定したパラメータの要件
- 重要なシステム・パラメータに対する値の計算。これらのパラメータについては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の AUTOGEN の節を参照。
- 新しいインストール・イメージ・リストの作成
- システムのページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズの計算
- 必要に応じて、システム・パラメータ値である、システムのページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズの値を調整する。
- システムのシャットダウンとリブート (オプション)

AUTOGEN の起動

AUTOGEN を起動するには、DCL プロンプトに次の形式でコマンドを入力します。

```
@SYS$UPDATE:AUTOGEN [開始フェーズ] [終了フェーズ] [実行モード]
```

開始フェーズ	AUTOGEN の実行が開始されるフェーズ。1.4.3 項「AUTOGEN のフェーズについて」に AUTOGEN フェーズのリストを示す。
終了フェーズ	AUTOGEN の実行が完了するフェーズ。1.4.3 項「AUTOGEN のフェーズについて」に AUTOGEN フェーズのリストを示す。
実行モード	次のいずれか <ul style="list-style-type: none"> • FEEDBACK フィードバックを使用する。 • NOFEEDBACK フィードバックを使用しない。 • CHECK_FEEDBACK フィードバックが有効であれば使用する。フィードバックが無効な場合は、無視するが、終了フェーズまで実行は続けられる。 • ブランク (実行モードが指定されない場合) フィードバックが有効であれば使用する。フィードバックが無効な場合は変更が行われる前に終了する。

AUTOGEN の起動と、指定できるコマンド行パラメータについての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』の AUTOGEN の節を参照してください。

AUTOGEN の動作の制御

表 1-1 「AUTOGEN の制御」に、AUTOGEN の動作を制御する方法を要約します。

表 1-1 AUTOGEN の制御

制御内容	使用する方法
AUTOGEN を実行させるための操作	AUTOGEN の起動時に開始フェーズと終了フェーズを指定する。
AUTOGEN によるパラメータ値の設定	AUTOGEN パラメータ・ファイル MODPARAMS.DAT に値を指定する。 使用しているハードウェア構成に関して AUTOGEN が正しい計算を行っているかどうか、AUTOGEN による計算結果を定期的に調べ、システム・パラメータ値が作業負荷の必要条件を確実に満たすようにする。値が適切でない場合は、MODPARAMS.DAT に必要な値を指定することにより調整する。MODPARAMS.DAT についての詳細は、1.4.4 項「AUTOGEN パラメータ・ファイル (MODPARAMS.DAT) について」を参照。
AUTOGEN によるフィードバック情報の使用	AUTOGEN の起動時に実行モードを指定する。 AUTOGEN により、稼働システムから収集した動的フィードバックを利用して、システムの性能を向上させることができる場合が多いが、フィードバック情報は無効な場合や不適切な場合もある。詳細は 1.4.1 項「AUTOGEN フィードバックについて」を参照。

1.4.1 AUTOGEN フィードバックについて

AUTOGEN フィードバックを利用すると、パラメータ値やシステム・ファイルのサイズを変更する回数が少なくてすみます。フィードバックにより、AUTOGEN は実際の作業負荷に基づいて、オペレーティング・システムのサイジングを自動的に行います。**サイジング**とは、システム資源(メモリおよびディスク空間)の割り当てを実際の作業負荷の必要条件に一致させることです。

フィードバックとは、オペレーティング・システム・エグゼクティブによって連続的に収集された、作業負荷を処理するためにシステムが使用するさまざまな資源に関する情報のことです。この情報は例外イベントが発生したときに収集されるため、収集作業はシステム性能に影響しません。AUTOGEN を**フィードバック・モード**で実行すると、AUTOGEN はこの情報を分析し、関連するすべてのパラメータ値を調整します。



注意:

構成を大幅に変更した後に AUTOGEN を実行するときには、**nofeedback** を指定して、必ず初期 AUTOGEN 設定を使用するようにしてください。1.4 項「AUTOGEN コマンド・プロセスについて」を参照してください。

AUTOGEN フィードバックは、次の資源に影響を与えます。影響を受けるシステム・パラメータについては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』の AUTOGEN の節を参照してください。

- 非ページング・プール
- ページング・プール
- ロック資源
- プロセスの数
- グローバル・ページ
- グローバル・セクション
- ファイル・システム・キャッシュ
- システム論理名テーブル・サイズ
- ページ・ファイル
- スワップ・ファイル

フィードバックは、AUTOGEN の SAVPARAMS フェーズで収集され、SYS\$SYSTEM:AGEN\$FEEDBACK.DAT ファイルに書き込まれます。このファイルはその後、

GETDATA フェーズに読み込まれます。AUTOGEN のフェーズについての詳細は、1.4.3 項「AUTOGEN のフェーズについて」を参照してください。

フィードバックが有効なのは、それがシステムの正常な作業負荷を反映している場合だけです。このため、AUTOGEN はフィードバックに対していくつかの基本的なチェックを行い、次のいずれかの条件に一致した場合には警告メッセージを出します。

- システムの連続稼働時間が 24 時間より少ない。
- フィードバックが作成されてから 30 日より多くの日数が経過している。

システムを変更した場合 (たとえばハードウェアのアップグレード、ユーザ数の変更、オプション製品のインストールを行った場合) には、しばらく新しいシステム環境で操作してから、再度 AUTOGEN を SAVPARAMS フェーズから実行してください。

VAX システムでは、フィードバックに必要な最小稼働時間を時間単位で指定するために、論理名 AGEN\$FEEDBACK_REQ_TIME を定義できます。詳細は 1.5.2 項「AUTOGEN フィードバックに必要な最小稼働時間の指定 (VAX のみ)」を参照してください。

AUTOGEN を実行すると、フィードバックが使用されているかどうかの次のように表示されます。

```
Feedback information was collected on 21-JAN-2000 14:00:08.53
Old values below are the parameter values at the time of collection.
The feedback data is based on 21 hours of up time.
Feedback information will be used in the subsequent calculations
```

1.4.2 フィードバック・レポート (AGEN\$PARAMS.REPORT) について

AUTOGEN によって算出されたシステム・パラメータの値とシステム・ファイルのサイズを使用するかどうか判断する必要があります。判断のための手助けとして、AUTOGEN は次の情報を含むレポート・ファイル (SYS\$SYSTEM:AGEN\$PARAMS.REPORT) を生成します。

- フィードバックから直接影響を受けるすべてのパラメータとシステム・ファイル
- 現在値
- 新しい値
- 各パラメータの計算に使用されるフィードバック
- MODPARAMS.DAT に対してユーザまたは弊社が行ったすべての変更内容
- AUTOGEN の実行中に表示されたすべてのアドバイス・メッセージと警告メッセージ
- VAX システムの場合、VMSPARAMS.DAT に対してユーザまたは弊社が行ったすべての変更内容
- Alpha システムおよび I64 システムの場合、GENPARAMS フェーズで検出されたパラメータ値

例 1-1 「AUTOGEN フィードバック・レポートの例」は、VAX システムの AUTOGEN フィードバック・レポートの例です。Alpha システムおよび I64 システムについては、フィードバック・レポートは似ていますが、この例とは同一ではありません。

情報メッセージの禁止

情報メッセージの表示を禁止するには、AGEN\$REPORT_NO_INFORMATIONALS 論理名を TRUE に定義します。メッセージは、AGEN\$REPORT_NO_INFORMATIONALS の値に関係なく、SYS\$SYSTEM:AGEN\$PARAMS.REPORT に入力されます。

ユーザ・レポートからの DCL 文のチェック

フィードバック・レポートには、MODPARAMS.DAT の DCL 文が含まれます。これらの DCL 文は、システム・パラメータや ADD_, MAX_, または MIN_ 拡張への単純な割り当てではありません。これらの文をレポートから除去するには、MODPARAMS.DAT の各文の先頭にドル記号 (\$) を付けます。

例 1-1 AUTOGEN フィードバック・レポートの例

AUTOGEN Parameter Calculation Report on node: NODE22
This information was generated at 23-APR-2000 01:45:47.87
AUTOGEN was run from GETDATA to TESTFILES using FEEDBACK

** No changes will be done by AUTOGEN **
The values given in this report are what AUTOGEN would
have set the parameters to.

Processing Parameter Data files

** WARNING ** - The system was up for less than 24 hours when the feedback
information was recorded. This could result in feedback information
that does not accurately reflect your typical work load.

Including parameters from: SYS\$SYSTEM:MODPARAMS.DAT

The following was detected within MODPARAMS.DAT
Please review immediately.

** INFORMATIONAL ** - Multiple MIN values found for MIN_CHANNELCNT.
Using MODPARAMS value (550) which is superseding OpenVMS value (255)

** INFORMATIONAL ** - Multiple MIN values found for MIN_SWPOUTPGCNT.
Using MODPARAMS value (1000) which is superseding OpenVMS value (500)

** INFORMATIONAL ** - Multiple MIN values found for MIN_PQL_DWSEXTENT.
Using MODPARAMS value (11000) which is superseding OpenVMS value (1024)

** INFORMATIONAL ** - Multiple MIN values found for MIN_PQL_MWSEXTENT.
Using MODPARAMS value (11000) which is superseding OpenVMS value (1024)

Feedback information was collected on 22-APR-2000 14:00:07.70
Old values below are the parameter values at the time of collection.
The feedback data is based on 13 hours of up time.
Feedback information will be used in the subsequent calculations

Parameter information follows:

MAXPROCESSCNT parameter information:
Feedback information.
Old value was 100, New value is 80
Maximum Observed Processes: 52

Information on VMS executable image Processing:

Processing SYS\$MANAGER:VMS\$IMAGES_MASTER.DAT

GBLPAGFIL parameter information:
Override Information - parameter calculation has been overridden.
The calculated value was 1024. The new value is 6024.
GBLPAGFIL has been increased by 5000.
GBLPAGFIL is not allowed to be less than 6024.

GBLPAGES parameter information:
Feedback information.
Old value was 43300, New value is 50000
Peak used GBLPAGES: 36622
Global buffer requirements: 6024

GBLSECTIONS parameter information:
Feedback information.
Old value was 400, New value is 400
Peak used GBLSECTIONS: 294
Override Information - parameter calculation has been overridden.

The calculated value was 350. The new value is 400.
GBLSECTIONS is not allowed to be less than 400.

LOCKIDTBL parameter information:

Feedback information.
Old value was 2943, New value is 3071
Current number of locks: 1853
Peak number of locks: 3200

LOCKIDTBL_MAX parameter information:

Feedback information.
Old value was 65535, New value is 65535

RESHASHTBL parameter information:

Feedback information.
Old value was 1024, New value is 1024
Current number of resources: 957

MSCP_LOAD parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 1. The new value is 0.
MSCP_LOAD has been disabled by a hard-coded value of 0.

MSCP_BUFFER parameter information:

Feedback information.
Old value was 128, New value is 128
MSCP server I/O rate: 0 I/Os per 10 sec.
I/Os that waited for buffer space: 0
I/Os that fragmented into multiple transfers: 0

SCSCONNCNT parameter information:

Feedback information.
Old value was 5, New value is 5
Peak number of nodes: 1
Number of CDT allocation failures: 0

SCSRESPCNT parameter information:

Feedback information.
Old value was 300, New value is 300
RDT stall count: 0

SCSBUFFCNT parameter information:

Feedback information.
Old value was 512, New value is 512
CIBDT stall count: 0

NPAGEDYN parameter information:

Feedback information.
Old value was 686592, New value is 783360
Maximum observed non-paged pool size: 815616 bytes.
Non-paged pool request rate: 47 requests per 10 sec.

LNMSHASHTBL parameter information:

Feedback information.
Old value was 1024, New value is 1024
Current number of shareable logical names: 1194

ACP_DIRCACHE parameter information:

Feedback information.
Old value was 88, New value is 88
Hit percentage: 99%
Attempt rate: 0 attempts per 10 sec.

ACP_DINDXCACHE parameter information:

Feedback information.
Old value was 25, New value is 25
Hit percentage: 97%
Attempt rate: 1 attempts per 10 sec.

ACP_HDRCACHE parameter information:

Feedback information.
Old value was 88, New value is 106

Hit percentage: 98%
Attempt rate: 17 attempts per 10 sec.

ACP_MAPCACHE parameter information:

Feedback information.
Old value was 8, New value is 8
Hit percentage: 2%
Attempt rate: 4 attempts per 10 sec.

PAGEDYN parameter information:

Feedback information.
Old value was 521728, New value is 542208
Current paged pool usage: 304160 bytes.
Paged pool request rate: 1 requests per 10 sec.

PFRATL parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 0. The new value is 1.
PFRATL has been disabled by a hard-coded value of 1.

WSDEC parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 35. The new value is 19.
WSDEC has been disabled by a hard-coded value of 19.

MPW_LOLIMIT parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 120. The new value is 2100.
MPW_LOLIMIT is not allowed to be less than 2100.

MPW_HILIMIT parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 1310. The new value is 4500.
MPW_HILIMIT is not allowed to be less than 4500.

LONGWAIT parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 30. The new value is 10.
LONGWAIT has been disabled by a hard-coded value of 10.

WSMAX parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 8200. The new value is 12000.
WSMAX is not allowed to be less than 12000.

PQL_DWSEXTENT parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 400. The new value is 11000.
PQL_DWSEXTENT is not allowed to be less than 11000.

PQL_DWSEXTENT parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 400. The new value is 11000.
PQL_DWSEXTENT is not allowed to be less than 11000.

PQL_MWSEXTENT parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 2048. The new value is 11000.
PQL_MWSEXTENT is not allowed to be less than 11000.

VAXCLUSTER parameter information:

Override Information - parameter calculation has been overridden.
The calculated value was 1. The new value is 0.
VAXCLUSTER has been disabled by a hard-coded value of 0.

Page, Swap, and Dump file calculations

Page and Swap file calculations.

PAGEFILE1_SIZE parameter information:

Feedback information.
Old value was 45200, New value is 50500

```

Maximum observed usage: 25265
PAGEFILE1_SIZE will be modified to hold 50500 blocks

PAGEFILE2_SIZE parameter information:
Feedback information.
  Old value was 154000, New value is 194400
  Maximum observed usage: 97175
PAGEFILE2_SIZE will be modified to hold 194400 blocks

** WARNING ** - The disk on which PAGEFILE2 resides would be
over 95% full if it were modified to hold 194400 blocks.
NODE22$DKA300:[SYSTEM_FILES]PAGEFILE.SYS will not be modified.
NODE22$DKA300:[SYSTEM_FILES]PAGEFILE.SYS will remain at 154002
blocks.

SWAPFILE1_SIZE parameter information:
Feedback information.
  Old value was 15000, New value is 15000
  Maximum observed usage: 14280
Override Information - parameter calculation has been overridden.
The calculated value was 21400. The new value is 15000.
SWAPFILE1_SIZE is not allowed to exceed 15000.
SWAPFILE1 will not be modified.

SWAPFILE2_SIZE parameter information:
Feedback information.
  Old value was 50000, New value is 26300
  Maximum observed usage: 1680
SWAPFILE2_SIZE will be modified to hold 26300 blocks

** WARNING ** - The disk on which SWAPFILE2 resides would be
over 95% full if it were modified to hold 26300 blocks.
NODE22$DKA300:[SYSTEM_FILES]SWAPFILE.SYS will not be modified.
NODE22$DKA300:[SYSTEM_FILES]SWAPFILE.SYS will remain at 50001 blocks.

Dumpfile calculations:

No dump file modifications would have been made.
Dumpfile will remain at 34116 blocks.

```

1.4.3 AUTOGEN のフェーズについて

AUTOGEN の起動時には、AUTOGEN に実行させる開始フェーズと終了フェーズを指定します。AUTOGEN は、指定された開始フェーズから終了フェーズまで、すべてのフェーズを実行します。開始フェーズと終了フェーズを指定すると、AUTOGEN は表 1-2 「AUTOGEN のフェーズ」に示す順序で各フェーズを実行します。

表 1-2 AUTOGEN のフェーズ

フェーズ	説明
SAVPARAMS	稼働システムの動的フィードバックを保存する。
GETDATA	AUTOGEN の計算で使用するすべてのデータを収集する。
GENPARAMS	新しいシステム・パラメータを生成し、インストールされたイメージ・リストを作成する。
TESTFILES	AUTOGEN が算出したページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズを表示する。このフェーズを開始フェーズにはできない。
GENFILES	必要に応じて新しいページ・ファイル、スワップ・ファイル、ダンプ・ファイルを生成する。開始フェーズにはできない。

表 1-2 AUTOGEN のフェーズ (続き)

フェーズ	説明
SETPARAMS	<p>SYSMAN を実行し、省略時のパラメータ・ファイルでの新しいシステム・パラメータの設定、オリジナルのパラメータの保存、新しいパラメータ・ファイル AUTOGEN.PAR の生成を行う。</p> <p>VAX システムの場合、省略時のパラメータ・ファイルは VAXVMSSYS.PAR。オリジナルのパラメータは VAXVMSSYS.OLD に保存される。</p> <p>Alpha システムの場合、省略時のパラメータ・ファイルは ALPHAVMSSYS.PAR。オリジナルのパラメータは ALPHAVMSSYS.OLD に保存される。</p> <p>164 システムの場合、省略時のパラメータ・ファイルは IA64VMSSYS.PAR。オリジナルのパラメータは IA64VMSSYS.OLD に保存される。</p>
SHUTDOWN	システムを手動でリブートできる状態にする。
REBOOT	自動的にシステムをシャットダウンし、リブートする。
HELP	画面にヘルプ情報を表示する。

AUTOGEN の各フェーズと各フェーズで影響を受けるファイルについての詳細は『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』の AUTOGEN の節を参照してください。

1.4.4 AUTOGEN パラメータ・ファイル (MODPARAMS.DAT) について

AUTOGEN は、GETDATA フェーズ中、MODPARAMS.DAT という名前のパラメータ・ファイルを読み込みます。このファイルにコマンドを追加すれば、AUTOGEN が設定するシステム・パラメータ値とファイル・サイズを制御できます。MODPARAMS.DAT を使用すれば、次のことができます。

操作	参照箇所
数値システム・パラメータの値を大きくする	1.5.1.1 項 「ADD_接頭辞を使って値を大きくする方法」
数値システム・パラメータの最小値を設定する	1.5.1.2 項 「MIN_接頭辞による最小値の指定方法」
数値システム・パラメータの最大値を設定する	1.5.1.3 項 「MAX_接頭辞による最大値の指定方法」
システム・パラメータの絶対値を指定する	1.5.1.4 項 「絶対値の指定方法」
外部のパラメータ・ファイルを取り込む	1.5.3 項 「外部パラメータ・ファイルの MODPARAMS.DAT への取り込み」
システム・ファイル (ページ・ファイル、スワップ・ファイル、ダンプ・ファイル) のサイズを指定する	2.16.1.2 項 「MODPARAMS.DAT でのページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズの制御」
VAXcluster ノードの数を定義する ¹	1.5.1.5 項 「VAXcluster ノード数の定義方法 (VAX のみ)」
イーサネット・アダプタの数を定義する ¹	1.5.1.6 項 「イーサネット・アダプタの数を定義する方法 (VAX のみ)」
メモリを追加する前にパラメータ値を設定する ¹	1.5.1.7 項 「メモリを追加する前にあらかじめパラメータ値を設定する方法 (VAX のみ)」
省略時の代替スタートアップ・コマンド・プロシージャを指定する	『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』

MODPARAMS.DAT に加えた変更内容を分かりやすく示すために、各行にコメントを追加します。コメントの先頭には感嘆符 (!) を付けます。



重要:

システム・パラメータとシステム・ファイルのサイズを変更する場合には、できるだけ、MODPARAMS.DAT ファイルにパラメータ設定値を指定する方法を使用してください。SYSMAN, SYSGEN, あるいは会話型ブートでシステム・パラメータ値またはファイル・サイズを変更しても、MODPARAMS.DAT に値を指定しなければ、AUTOGEN は次回の実行時にパラメータ値とファイル・サイズを再計算します。詳細は 1.5.1 項「MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御」を参照してください。

例

次に MODPARAMS.DAT ファイルの例を示します。

```
!
! ***** A Sample MODPARAMS.DAT for Node NODE22 *****
!
! MODPARAMS.DAT for "NODE22"
! REVISED: 04/29/00 -CHG- Upped GBLPAGES to account for ADA.
!
SCSNODE          = "NODE22"          ! This is not calculated by AUTOGEN.
SCSSYSTEMID      = 19577             ! This is not calculated by AUTOGEN.
TTY_DEFCHAR2     = %X0D34           ! This is not calculated by AUTOGEN.
ADD_ACP_DIRCACHE= 150               ! Hit rate was only 65% on directory cache.
MIN_PAGEDYN      = 500000           ! PAGEDYN must be at least 1/2 Mbyte to
! account for a large number of logical names.
!
MAX_PAGEFILE1_SIZE = 15000          ! Maximum size for primary page.
MAX_SWAPFILE       = 5000           ! Maximum size for swap file space.
MAX_DUMPFIL       = 32768          ! Maximum size for dump file space.
!
ADD_GBLPAGES      = 425+507+157     ! Account for MCS, BLISS32 and ADA.
ADD_GBLSECTIONS  = 4 + 5 + 2       ! Account for MCS, BLISS32 and ADA.
VIRTUALPAGECNT   = 144264          ! So that we can read MONSTR's 68Mb dumps.
!
! end of MODPARAMS.DAT for NODE22
```

1.5 AUTOGEN によるシステム・パラメータの変更

システム・パラメータを変更する場合には、できるだけ次に示すように、AUTOGEN を 2 段階に分けて実行してください。

1. 第 1 段階では、次のコマンドを使って AUTOGEN を実行する。

```
$ @SYS$UPDATE:AUTOGEN SAVPARAMS TESTFILES
```

このコマンドを実行すると、AUTOGEN は次の作業を行う。

- 現在のフィードバックの保存
- 計算に必要なすべての情報の収集
- システム・パラメータ値の計算
- フィードバック・レポートの生成
- SETPARAMS.DAT への情報の書き込み

計算の元となるデータ (PARAMS.DAT), 計算結果 (SETPARAMS.DAT), および生成されたレポート (AGEN\$PARAMS.REPORT) を調べる。

パラメータの設定内容に満足できない場合には、MODPARAMS.DAT を編集してパラメータ値を変更する。編集方法については、1.5.1 項「MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御」を参照。パラメータ値を変更したら、ファイル・サイズを変更

したい場合、2.16 項「ページ・ファイル、スワップ・ファイル、ダンプ・ファイルの作成と変更」を参照してサイズを変更する。その後、GETDATA フェーズから AUTOGEN を実行する。

SETPARAMS.DAT の内容に満足した場合には、ステップ 2 に進む。

2. 第 2 段階では、次のコマンドを使って、AUTOGEN をもう一度実行する。

```
$ @SYS$UPDATE:AUTOGEN GENPARAMS REBOOT
```

この AUTOGEN コマンドは、SYSMAN を実行して新しいシステム・パラメータ値を更新し、リポート時にそれらのパラメータを使用してシステムをブートする。この方法では、システム・ファイルは変更されない点に注意。

1.5.1 MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御

AGEN\$PARAMS.REPORT ファイルおよび SETPARAMS.DAT ファイルを調べた結果、ハードウェア構成データを修正したり、AUTOGEN が選択したシステム・パラメータ値を変更することになった場合は、この項で説明しているように、MODPARAMS.DAT ファイルを編集してパラメータ値を指定してください。



重要:

パラメータの値を指定するためには、PARAMS.DAT ではなく、必ず MODPARAMS.DAT を使用してください。PARAMS.DAT の内容を変更すると、AUTOGEN が正しく機能しない場合があります。

MODPARAMS.DAT ファイルを使ってページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズを制御する方法については、2.16.1.2 項「MODPARAMS.DAT でのページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズの制御」を参照してください。

MODPARAMS.DAT ファイル中で、次の形式でシンボルを定義すると、パラメータ値を制御することができます。

制御方法	シンボルの形式	参照箇所
指定された量だけ値を大きくする	ADD_*	1.5.1.1 項「ADD_ 接頭辞を使って値を大きくする方法」
最小値を指定する	MIN_*	1.5.1.2 項「MIN_ 接頭辞による最小値の指定方法」
最大値を指定する	MAX_*	1.5.1.3 項「MAX_ 接頭辞による最大値の指定方法」
絶対値を指定する	パラメータ名	1.5.1.4 項「絶対値の指定方法」

MODPARAMS.DAT を使ってシンボルを定義する場合には、次のことを確認してください。

- 値が正しく、そのパラメータに対して有効である。桁数は適切で、コンマは使用されていない。
- そのシンボルが MODPARAMS.DAT 内で一度だけ定義されている。
- シンボルの値がコメントになっていない。
- シンボル名のスペルが正しく完全である (省略されていない)。
- 文字列の値が二重引用符 ("") 中にある。



重要:

MODPARAMS.DAT や他のパラメータ・ファイルを読み込む場合、AUTOGEN はファイルに指定されたシンボル名が有効かどうかチェックします。有効でない場合、AUTOGEN は AGEN\$PARAMS.REPORT に警告メッセージを書き込みます。しかし、AUTOGEN がチェックするのはシンボル名だけで、そのシンボルに指定された値が有効かどうかはチェックしません。

値が無効でもその行が無視されることはありません。AUTOGEN は指定された値の使用を試みます。

シンボルが指定されている行に等号記号(=)以外の DCL の命令文が含まれていると、そのシンボルはチェックされません。たとえば、DCL の IF 文がある行に指定されているシンボル名は、有効かどうかはチェックされません。その場合、AUTOGEN は AGEN\$PARAMS.REPORT に警告メッセージを書き込みます。

MODPARAMS.DAT に加えた変更内容を分かりやすく示すために、ファイルを変更するたびに必ず各行にコメントを追加します。コメントの先頭には感嘆符 (!) を付けます。

1.5.1.1 ADD_ 接頭辞を使って値を大きくする方法

ADD_ 接頭辞を使って数値パラメータの値を大きくします。新しい値は GENPARAMS フェーズの以降の AUTOGEN 計算によって更新されます。次に、ADD_ 接頭辞の使用例を示します。

```
ADD_GBLPAGES=500
ADD_NPAGEDYN=10000
```

AUTOGEN が計算するパラメータの場合は、AUTOGEN の計算結果に加算されます。AUTOGEN が計算しないパラメータの場合には、現在値ではなく、省略時の値に加算されます。なお、AUTOGEN により影響を受けるパラメータの表が『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』の AUTOGEN の節にありますので参照してください。



注意:

ADD_ で指定された値は計算された値に一度だけ加えられます。フィードバック計算のための連続実行は、累積されません。

通常、フィードバック機構によって計算されたパラメータを変更するときに、ADD_ 接頭辞を使用することはありません。それは、フィードバックの結果が作業負荷を正確に反映しなければならぬためです。もし、フィードバックで ADD_ 接頭辞を使用する場合は、AUTOGEN が SETPARAMS フェーズ以降まで実行されても、AUTOGEN は値を一度しか加算しない点に注意してください。AUTOGEN の計算値より上に最小値を設定したい場合には、MIN_ 接頭辞を使用します。

1.5.1.2 MIN_ 接頭辞による最小値の指定方法

AUTOGEN による設定値が指定した値より小さくならないようにするためには、MIN_ 接頭辞を使用します。MIN_ は、AUTOGEN が設定できるパラメータの最小値を示します。最小値を 400,000 に設定する例を次に示します。

```
MIN_PAGEDYN = 400000
```

1.5.1.3 MAX_ 接頭辞による最大値の指定方法

AUTOGEN が、指定した値より大きな値にパラメータを設定しないようにするためには、MAX_ 接頭辞を使用します。MAX_ は、AUTOGEN によって設定できるパラメータの最大値を示します。最大値を 400,000 に設定する例を次に示します。

```
MAX_PAGEDYN = 400000
```

1.5.1.4 絶対値の指定方法

この方法は、AUTOGENが計算しないパラメータの値を指定する場合に使用します。AUTOGENの計算で変更されるシステム・パラメータの表が『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』の AUTOGEN の節にありますので参照してください。



注意:

この方法は、システム環境を記述するパラメータ (SCSNODE や SCSSYSTEMID など) に限って使用してください。AUTOGEN により計算されるパラメータでこの方法を指定すると、AUTOGEN の計算ができなくなります。絶対値を指定する代わりに、次のいずれかの方法を用いてください。

- MIN_ 接頭辞を使用して最小値を指定する
- MAX_ 接頭辞を使用して最大値を指定する
- ADD_ 接頭辞を使用して値を大きくする

パラメータの絶対値を指定するためには、MODPARAMS.DAT に次の形式で代入文を追加します。

パラメータ名 = パラメータ値 ! comment

たとえば、次のコマンドはノード名 BIGVAX を SCSNODE パラメータに代入します。

```
SCSNODE = "BIGVAX"    ! the node name
```



注意:

文字列の値は、二重引用符 (" ") で囲む必要があります。

1.5.1.5 VAXcluster ノード数の定義方法 (VAX のみ)

VAXcluster 環境では、NUM_NODES シンボルを使用して、VAXcluster のメンバが一時的に変化しても AUTOGEN による VAXcluster に関連するパラメータの値の計算に影響しないようにすることができます。具体的には、MODPARAMS.DAT で NUM_NODES シンボルを定義して、VAXcluster で稼働させるノードの数を指定します。AUTOGEN はこの値を使用して、VAXcluster ノードの数に影響されるパラメータを設定します。たとえば、MODPARAMS.DAT に次の行を追加します。

```
NUM_NODES = 30
```

1.5.1.6 イーサネット・アダプタの数を定義する方法 (VAX のみ)

VAXcluster 環境では、NUM_ETHERADAPT シンボルを MODPARAMS.DAT に定義して、VAXcluster 内のイーサネット・アダプタの合計数を指定します。たとえば、MODPARAMS.DAT に次の行を追加します。

```
NUM_ETHERADAPT = 40
```

1.5.1.7 メモリを追加する前にあらかじめパラメータ値を設定する方法 (VAX のみ)

VAX システムでは、大容量のメモリ (512 MB 以上) を追加することにより、システム・ハードウェアをアップグレードしようとする場合、システム・パラメータを追加分のメモリに適するようあらかじめ設定しておくようにします。システム・パラメータをあらかじめ設定しておくことにより、不適切なパラメータ値を指定することで発生するメモリのアップグレードに関する問題を、最小限に抑えることができます。

作業方法

次の手順を実行してください。

1. SYS\$SYSTEM:MODPARAMS.DAT に次の形式の 1 行を加える。
MEMSIZE = アップグレード後のメモリの総ページ数
次に例を示す。

```
MEMSIZE = 2048 * 1024 ! (2048 page per MB * 1GB of memory)
```

2. SETPARAMS フェーズで AUTOGEN を実行する。
3. ハードウェアのアップグレードを実行して、追加メモリを加える。
4. MODPARAMS.DAT を編集して、ステップ 1 で追加した行を削除する。

1.5.1.8 DECnet に関連するパラメータを上書きする方法

DECnet の存在 (または不在) に関する AUTOGEN の観測を上書きするには、MODPARAMS.DAT のパラメータ LOAD_DECNET_IMAGES を TRUE (または FALSE) に設定します。同期ネットワーク・ハードウェアがないシステムにおいて非同期 DECnet を実行したい場合には、設定値を制御すると役に立ちます。

1.5.1.9 NPAGEDYN と NPAGEVIR の値の設定

物理メモリが多くない OpenVMS VAX システムのために、AUTOGEN は NPAGEDYN が物理メモリの 10% を超えた時、あるいは NPAGEVIR が物理メモリの 33% を超えた時に、警告メッセージをレポートに記録します。

AUTOGEN は計算値を、NPAGEDYN は物理メモリの 20% に NPAGEVIR は物理メモリの 50% に制限しています。これらの計算値は物理メモリが 16MB 以下のほとんどのシステムで妥当な値です。システムでこれ以上大きな値を設定する必要がある場合、MODPARAMS.DAT に書き込むことによって AUTOGEN の計算値を上書きすることができます。

1.5.2 AUTOGEN フィードバックに必要な最小稼働時間の指定 (VAX のみ)

VAX システムでは、システムの稼働時間がシステムの正常な作業負荷を十分に反映していなければ、AUTOGEN フィードバックは有効ではありません。省略時の設定では、AUTOGEN はデータが作成されてから 24 時間より多く経過しないとフィードバックを使用しません。VAX システムでは、論理名 AGEN\$FEEDBACK_REQ_TIME を定義することにより、省略時の設定と異なる最小稼働時間(データの生成後、フィードバックに利用できない期間)を時間単位で指定できます。

たとえば、論理名を次のように定義した場合、AUTOGEN はデータが生成されてから 19 時間より多く経過しないとフィードバック・データを使用しません。

```
$ DEFINE/SYSTEM AGEN$FEEDBACK_REQ_TIME 19
```

システムが起動するたびにこの論理名を定義するためには、このコマンドを SYLOGICALS.COM に追加します。

1.5.3 外部パラメータ・ファイルの MODPARAMS.DAT への取り込み

外部のパラメータ・ファイルを MODPARAMS.DAT に取り込むことができます。たとえば、あるシステム・パラメータを VAXcluster または OpenVMS Cluster 環境内のすべてのノードと同じ値に設定すると同時に、他のシステム・パラメータをノード固有の値にしたいことがあります。そのような場合は、クラスタ共通の値を別のファイルで指定し、クラスタ内の各システムの MODPARAMS.DAT にそのファイルを取り込むようにします。

パラメータ・ファイルを取り込むためには、MODPARAMS.DAT、あるいは MODPARAMS.DAT に取り込まれる任意のパラメータ・ファイルに次の形式でコマンドを追加します。

AGEN\$INCLUDE_PARAMS 完全なディレクトリ指定:ファイル名

例

CLUSTERPARAMS.DAT という名前のクラスタ共通のパラメータ・ファイルを取り込む場合には、まず、次の名前の共通のパラメータ・ファイルを作成します。

```
SYS$COMMON:[SYSEXEC]CLUSTERPARAMS.DAT
```

次に、各クラスタのシステム固有のディレクトリの MODPARAMS.DAT ファイルに次の行を追加します。

```
AGEN$INCLUDE_PARAMS SYS$COMMON:[SYSEXEC] CLUSTERPARAMS.DAT
```

1.5.4 DCL 文のログの停止

MODPARAMS.DAT の内容は、DCL 文として評価されます。したがって、シンボルにシステム・パラメータではない名前 (たとえば、他の値に基づくスクラッチ変数または条件付き割り当て) を割り当てることができます。通常、そのような割り当てはすべて AGEN\$PARAMS.REPORT にログが取られ、ログが取られた多くの文によって必要のない大きなファイルが作成されていきます。

このような割り当てすべてにドル記号 (\$) を接頭辞として付けることにより、AGEN\$PARAMS.REPORT にログを取られたくない割り当てを指定することができます。AUTOGEN は、MODPARAMS.DAT 中のドル記号で始まるレコードを検出すると、既知のシステム・パラメータのリストをチェックしないで、AGEN\$PARAMS.REPORT にこのレコードのログを取りません。

1.6 AUTOGEN レポートの自動化

バッチ・モードのコマンド・プロシージャを作成して、AUTOGEN を定期的に自動実行させ、その結果得られるフィードバック・レポートを適切な Mail アカウントに送信しておくようにしてください。例 1-2 「AUTOGEN コマンド・プロシージャの例」は、このようなコマンド・プロシージャの例です。



注意:

このコマンド・プロシージャでは、システム・パラメータの適切な値を計算してレポートを送信するためにだけ AUTOGEN を実行しています。AUTOGEN の実行によりシステム・パラメータを変更したり、システムをリブートすることはありません。レポートを調べた結果、システム・パラメータを変更することになった場合には、1.6.1 項「AUTOGEN レポートを調べた後のパラメータ値の変更」にある指示に従ってください。

例 1-2 「AUTOGEN コマンド・プロシージャの例」に示したコマンド・プロシージャは、AUTOGEN を 2 段階に分けて実行します。第 1 段階では、AUTOGEN を作業負荷がピークになる時間帯に実行し、現実的なシステムの作業負荷のデータを収集します。この段階ではシステムの性能が低下することはありません。第 2 段階では、AUTOGEN をシステムの作業負荷が軽い時間帯に実行し、第 1 段階で収集したデータを解釈します。

このプロシージャで、ファイル AGEN\$PARAMS.REPORT に含まれている結果のレポートが SYSTEM アカウントに送信されます。このレポートを定期的に検討し、システムの負荷が変更されているかどうかを調べてください。

例 1-2 「AUTOGEN コマンド・プロシージャの例」に、コマンド・プロシージャの例を示します。このプロシージャを参考にして、使用中のシステム構成に合ったコマンド・プロシージャを作成してください。

例 1-2 AUTOGEN コマンド・プロシージャの例

```
$ BEGIN$:      ! ++++++++ AGEN_BATCH.COM ++++++++
$  on warning then goto error$
$  on control_y then goto error$

$!
$! Setup process
$!
$! Set process information
$  set process/priv=all/name="AUTOGEN Batch"
$! Keep log files to a reasonable amount
$  purge/keep=5 AGEN_Batch.log
$  time = f$time()                ! Fetch current time
$  hour = f$integer(f$cvtime(time,, "hour") )    ! Get hour
$  today = f$cvtime(time,, "WEEKDAY")           ! Get Day of the week
$  if f$integer(f$cvtime(time,, "minute") ) .ge. 30 then hour = hour + 1

$!
$! Start of working day...
$!
$ 1AM$:
$  if hour .le. 2
$    then
$      next_time = "today+0-14"
$      gosub submit$                ! Resubmit yourself
$      set noon
$!

$!      Run AUTOGEN to TESTFILES using the parameter values collected earlier

$!      in the day (i.e., yesterday at 2:00pm)
$  if today .eqs. "Tuesday" .OR. today .eqs. "Thursday" .OR. -
$    today .eqs. "Saturday"
$  then
$    @sys$update:autogen GETDATA TESTFILES feedback 1
$    mail/sub="AUTOGEN Feedback Report for system-name" -
$  sys$system:agen$params.report system 2
$    ! Clean up
$    purge/keep=7 sys$system:agen$feedback.report 3
$    purge/keep=7 sys$system:agen$feedback.dat
$    purge/keep=7 sys$system:params.dat
$    purge/keep=7 sys$system:autogen.par
$    purge/keep=7 sys$system:setparams.dat
$    purge/keep=7 sys$system:agen$addhistory.tmp
$    purge/keep=7 sys$system:agen$addhistory.dat
$  endif
$  goto end$
$  endif

$!
$ 2PM$:
$  if hour .le. 15
$    then
$      next_time = "today+0-17"
$      gosub submit$
$      if today .eqs. "Monday" .OR. today .eqs. "Wednesday" .OR. -
$        today .eqs. "Friday"
$      then
$        @sys$update:autogen SAVPARAMS SAVPARAMS feedback 4
$      endif
$    goto end$
$  endif

$!
```

```

$ 5PM$:
$   if hour .le. 18
$     then
$       next_time = "tomorrow+0-1"
$       gosub submit$
$     endif

$!
$! End of working day...
$!
$ END$:      ! ----- BATCH.COM -----
$   exit
$!++
$! Subroutines
$!--
$!
$ SUBMIT$:
$   submit/name="AGEN_Batch"/restart/noprint - 5
$     /log=AGEN_batch.log -
$     /queue=sys$batch/after="'next_time'" sys$system:AGEN_batch.com
$   return
$!++
$! Error handler
$!--

$ ERROR$:
$   mail/sub="AGEN_BATCH.COM - Procedure failed." _nl: system
$   goto end$

```

次に、このコマンド・プロシージャで実行されるタスクを説明します。

- 1 作業負荷が軽い時間帯に AUTOGEN の第 2 段階を実行し、第 1 段階で収集したデータを解釈する。
- 2 AGEN\$PARAMS.REPORT という名前の結果のレポート・ファイルを SYSTEM アカウントにメールする。
- 3 作成したファイルを削除する。
- 4 作業負荷がピークになる時間帯に AUTOGEN の第 1 段階を実行し、実際の作業負荷のデータを収集する。このコマンドは非常に高速なイメージを実行するため、システムの応答性を低下させることはない。
- 5 コマンド・プロシージャを再びキューに登録する。

1.6.1 AUTOGEN レポートを調べた後のパラメータ値の変更

1.6 項「AUTOGEN レポートの自動化」で説明したコマンド・プロシージャのレポートにおいて、AUTOGEN の計算結果が現在値と異なっている場合には、次に示すいずれかの方法で、AUTOGEN の設定内容を修正してください。

- ただちにシステムをシャットダウンしてリブートできる場合には、次のコマンドを実行する。

```
$ @SYS$UPDATE:AUTOGEN GETDATA REBOOT FEEDBACK
```

- ただちにシステムのシャットダウンとリブートを行うことができない場合には、次のコマンドを実行してシステム・パラメータを再設定する。

```
$ @SYS$UPDATE:AUTOGEN GETDATA SETPARAMS FEEDBACK
```

新しいパラメータは、システムのブート後、有効になる。

1.7 SYSMAN ユーティリティによるシステム・パラメータの管理



注意:

システム・パラメータを変更する場合には、できるだけ AUTOGEN を使用してください。詳細は1.5 項「AUTOGEN によるシステム・パラメータの変更」を参照してください。また、ノード・グループに関するシステム・パラメータを表示したり、パラメータを一時的に変更したい場合には、SYSMAN を使用してください。

システム管理ユーティリティ (SYSMAN) を使用すると、1 つのシステムだけではなく、クラスター全体、あるいはノードの任意のグループに対応するシステム・パラメータを調べたり、変更することができます。SYSMAN で利用できる PARAMETERS コマンドは、OpenVMS のシステム生成ユーティリティ (SYSGEN) のパラメータ設定機能を持っています。

次に、SYSMAN ユーティリティを使ってシステム・パラメータを管理できる作業を示します。

作業	参照箇所
パラメータ値の表示	1.7.2 項「SYSMAN によるパラメータ値の表示」
パラメータ・ファイル内の現在値の変更	1.7.3 項「SYSMAN によるパラメータ・ファイルの変更」
稼働中のシステムのアクティブ値の変更 ¹	1.7.4 項「SYSMAN によるアクティブ値の変更」

1 ダイナミック・システム・パラメータだけに適用

表 1-3 「SYSMAN PARAMETERS コマンド」に、SYSMAN が提供するコマンドと機能を示します。

表 1-3 SYSMAN PARAMETERS コマンド

コマンド	機能
PARAMETERS SHOW	パラメータ値を表示する。
PARAMETERS USE	パラメータのセットを、メモリまたはディスクから調査または変更のための一時作業領域に読み込む。ファイル名、または追加パラメータ ACTIVE か CURRENT が必要。
PARAMETERS SET	パラメータ値を変更する。変更内容は作業領域内でのみ有効。変更内容をより永久的なものにするためには、PARAMETERS WRITE コマンドが必要。
PARAMETERS WRITE	作業領域の内容をメモリまたはディスクに書き込む。ファイル名、または追加パラメータ ACTIVE か CURRENT が必要。

一時作業領域についての詳細は、1.7.1 項「パラメータ値と SYSMAN について」を参照してください。

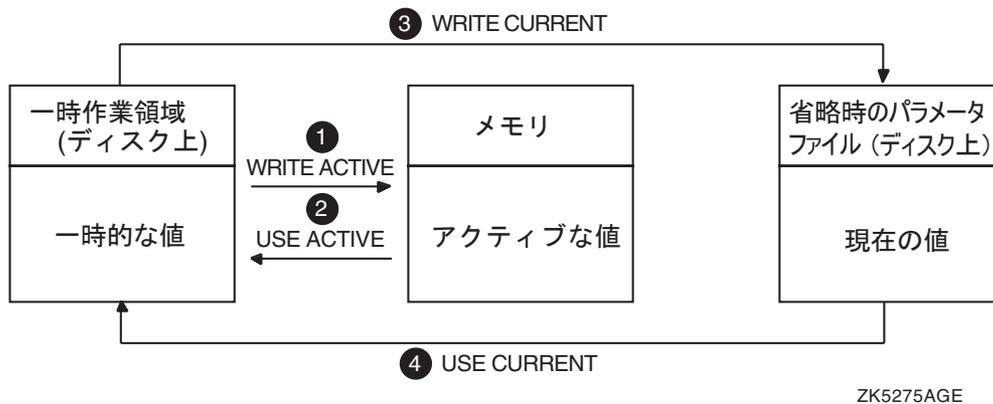
1.7.1 パラメータ値と SYSMAN について

1.1.1 項「省略時の値、現在値、アクティブ値」で説明しているように、システム・パラメータの値にはいくつかの種類があります。簡単にまとめると**現在値**とは、ディスク上の省略時のパラメータ・ファイルに格納されている値のことです。**アクティブ値**とは、メモリ内に格納され、システムの稼働中に使用される値のことです。これらの値のほかにも、SYSMAN はディスク上の独自の作業領域にパラメータ値を一時的に書き込みます。この値を一時値と呼びます。図 1-2 「SYSMAN パラメータの一時値、アクティブ値、現在値」は、この 3 種類の値と、SYSMAN コマンドがそれらの値にどう影響するかを示しています。次に、図中の動作を説明します。

1. WRITE ACTIVE は、一時的パラメータ値をメモリに書き込む。
2. USE ACTIVE は、値をメモリから作業領域に読み込む。ユーザはここで値を変更することができます。

- WRITE CURRENT は、一時的パラメータ値をディスクに書き込む。書き込まれた値が現在の値となる。次にシステムをブートしたとき、これらの値がアクティブになる。
- USE CURRENT は、現在の値をディスクから作業領域に読み込む。ユーザはここで値を変更することができる。

図 1-2 SYSMAN パラメータの一時値、アクティブ値、現在値



セッション中にパラメータ値を表示または変更する場合、通常のセッションでは次の手順に従ってください。

- USE コマンドを使って、パラメータの値を SYSMAN の一時作業領域に読み込む。アクティブ値を読み込むときには USE ACTIVE を、現在値を読み込むときには USE CURRENT をそれぞれ使用する。
- SHOW コマンドを使って、パラメータ値を表示する。
- SET コマンドを使って、パラメータ値を変更する。変更した値を有効にするには、WRITE コマンドを実行しなければならない。
- WRITE コマンドを使って、変更内容を有効にする。
 - WRITE ACTIVE を実行すると、パラメータの値はアクティブ値に書き込まれる。ただし、アクティブ値を変更できるのはダイナミック・パラメータの場合だけである。
 - WRITE CURRENT を実行すると、パラメータの値は現在値に書き込まれる。

システム・パラメータについての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

1.7.2 SYSMAN によるパラメータ値の表示

SYSMAN の PARAMETERS SHOW コマンドを使用すると、クラスタ内のすべてのノードのパラメータ値を表示することができます。

例

- パラメータに関する情報の表示方法を示す。/LGI 修飾子を使用すると、すべてのログイン・セキュリティ制御パラメータを表示する。その他にも、/ACP, /ALL, /SPECIAL などのパラメータを表示できる。パラメータとパラメータ・カテゴリについては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> PARAMETERS SHOW/LGI
```

```
Parameters in use: Active
Parameter Name          Current   Default   Min.      Max.      Unit      Dynamic
-----
LGI_BRK_TERM            0         1         0         1         Boolean   D
LGI_BRK_DISUSER         0         0         0         1         Boolean   D
LGI_PWD_TMO              30        30        0         255      Seconds   D
LGI_RETRY_LIM            3         3         0         255      Tries     D
LGI_RETRY_TMO            20        20        0         255      Seconds   D
```

LGI_BRK_LIM	5	5	0	255 Failures	D
LGI_BRK_TMO	300	300	0	-1 Seconds	D
LGI_HID_TIM	300	300	0	-1 Seconds	D

2. 次の例では、SYSMAN を起動し、NODE21 および NODE22 から構成されるローカル・クラスタを環境として指定する。また、ユーザ、ターミナル、ノードのログインを試みる時間 (秒数) を制御する LGI_BRK_TMO パラメータのアクティブ値を表示する。この例では、アクティブ値は 600 である。

```

$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> SET ENVIRONMENT/CLUSTER
%SYSMAN-I-ENV, Current command environment:
  Clusterwide on local cluster
  Username MORIN will be used on nonlocal nodes
SYSMAN> PARAMETERS SHOW LGI_BRK_TMO
Node NODE21: Parameters in use: ACTIVE
Parameter Name      Current  Default  Minimum  Maximum Unit  Dynamic
-----
LGI_BRK_TMO         600     300      0        -1 Seconds  D

Node NODE22: Parameters in use: ACTIVE
Parameter Name      Current  Default  Minimum  Maximum Unit  Dynamic
-----
LGI_BRK_TMO         600     300      0        -1 Seconds  D

```

1.7.3 SYSMAN によるパラメータ・ファイルの変更

SYSMAN の PARAMETERS WRITE コマンドを使用すると、システム・パラメータ値およびシステム固有のスタートアップ・コマンド・プロシージャを、ユーザが選択したパラメータ・ファイル、ディスク上の現在のシステム・パラメータ・ファイルに書き込むことができます。

DCL の SET MESSAGE コマンドを使ってシステム・メッセージの形式を変更していない限り、PARAMETERS WRITE CURRENT コマンドを実行すると、メッセージが OPCOM に送信され、イベントが記録されます。



注意:

PARAMETERS WRITE CURRENT コマンドを実行すると、現在操作している値だけでなく、パラメータのすべてのアクティブ値または現在値がディスクに書き込まれます。

例

1. 新しいパラメータ指定ファイルを作成する。

```
SYSMAN> PARAMETERS WRITE SYS$SYSTEM:NEWPARAM
```

2. PARAMETERS SET コマンドに続けて PARAMETERS WRITE コマンドを実行すると、ディスク上の現在のパラメータ・ファイルが変更される。

```
SYSMAN> PARAMETERS SET LGI_BRK_TMO 300
SYSMAN> PARAMETERS WRITE CURRENT
```

1.7.4 SYSMAN によるアクティブ値の変更

SYSMAN の PARAMETERS SET コマンド、PARAMETERS WRITE コマンド、および PARAMETERS USE コマンドを使用すると、アクティブ・パラメータ値を変更できます。

アクティブ値を変更するとメモリ上の値が変更されるため、ただちにダイナミック・パラメータに影響します。ダイナミック・パラメータについての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。また、SYSMAN の PARAMETERS SHOW/DYNAMIC コマンドを使っても調べることができます。ダイナミック・パラメータ以外のパラメータ値は、システムの稼働中には変更できません。

アクティブ値を変更しても、ディスク上のシステム・パラメータ・ファイルには影響はありません。次のシステム・ブート時には、以前の現在値がアクティブ値として使用されるためです。

新しいアクティブ・パラメータ値を設定し、その値を以降のブート操作で使用する場合には、PARAMETERS WRITE CURRENT コマンドを使って、新しい値を現在のパラメータ・ファイルに書き込みます。次の例を参照してください。



重要:

SYSMAN によって変更されたパラメータ値は、AUTOGEN コマンド・プロシージャによって上書きされます。SYSMAN で行った変更内容を保持するためには、SYS\$SYSTEM:MODPARAMS.DAT に新しいパラメータ値を指定する必要があります。手順については、1.5.1 項「MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御」を参照してください。

例

1. LGI_BRK_TMO の値を作業領域で 300 に変更し、それをアクティブ値としてメモリに書き込んだ後、アクティブ値を表示する。

```
SYSMAN> PARAMETERS SET LGI_BRK_TMO 300

SYSMAN> PARAMETERS WRITE ACTIVE
SYSMAN> PARAMETERS SHOW LGI_BRK_TMO
Node NODE21: Parameters in use: ACTIVE
Parameter Name      Current  Default  Minimum  Maximum Unit  Dynamic
-----
LGI_BRK_TMO         300     300      0        -1 Seconds  D

Node NODE22: Parameters in use: ACTIVE
Parameter Name      Current  Default  Minimum  Maximum Unit  Dynamic
-----
LGI_BRK_TMO         300     300      0        -1 Seconds  D
```

2. LGI_BRK_TMO に含まれている現在のパラメータ値をディスクから作業領域に呼び出した後、LGI_BRK_TMO を表示する。この例では、ディスク上の現在値は 600 である。

```
SYSMAN> PARAMETERS USE CURRENT
SYSMAN> PARAMETERS SHOW LGI_BRK_TMO

Node NODE21: Parameters in use: CURRENT
Parameter Name      Current  Default  Minimum  Maximum Unit  Dynamic
-----
LGI_BRK_TMO         600     300      0        -1 Seconds  D

Node NODE22: Parameters in use: CURRENT
Parameter Name      Current  Default  Minimum  Maximum Unit  Dynamic
-----
LGI_BRK_TMO         600     300      0        -1 Seconds  D
```

3. LGI_BRK_TMO の値 (600) を作業領域からメモリに書き込む。書き込まれた値は、稼働中のシステムのアクティブ値となる。PARAMETER WRITE ACTIVE コマンドにより、LGI_BRK_TMO だけではなく、すべてのパラメータ値が作業領域からメモリに書き込まれる点に注意。

```
SYSMAN> PARAMETERS WRITE ACTIVE

SYSMAN> PARAMETERS USE ACTIVE

SYSMAN> PARAMETERS SHOW LGI_BRK_TMO

Node NODE21: Parameters in use: ACTIVE
```

Parameter Name	Current	Default	Minimum	Maximum	Unit	Dynamic
LGI_BRK_TMO	600	300	0	-1	Seconds	D

Node NODE22: Parameters in use: ACTIVE

Parameter Name	Current	Default	Minimum	Maximum	Unit	Dynamic
LGI_BRK_TMO	600	300	0	-1	Seconds	D

1.8 SYSGEN ユーティリティによるシステム・パラメータの管理



注意:

システム・パラメータを変更する場合には、できるだけ AUTOGEN を使用してください (詳細は 1.5 項「AUTOGEN によるシステム・パラメータの変更」を参照)。何らかの理由で AUTOGEN を使用できない場合には、SYSMAN ユーティリティを使用してください (詳細は 1.7 項「SYSMAN ユーティリティによるシステム・パラメータの管理」を参照)。

次に、SYSGEN ユーティリティを使ってシステム・パラメータを管理できる作業を示します。ただし、この方法はなるべく使用しないでください。

作業	参照箇所
パラメータ値の表示	1.8.2 項「SYSGEN によるパラメータ値の表示」
省略時のパラメータ・ファイル内の現在値の変更	1.8.3 項「SYSGEN によるシステム・パラメータ・ファイルの変更」
稼働中のシステムのアクティブ値の変更 ¹	1.8.4 項「SYSGEN によるアクティブ値の変更」
新しいパラメータ・ファイルの作成	1.8.5 項「SYSGEN による新規パラメータ・ファイルの作成」

¹ ダイナミック・システム・パラメータのみ。

SYSGEN でシステム・パラメータを管理するときに使用するコマンドを、表 1-4 「システム・パラメータとともに使用される SYSGEN コマンド」に示します。SYSGEN コマンドについての詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』の SYSGEN の節を参照してください。

表 1-4 システム・パラメータとともに使用される SYSGEN コマンド

コマンド	機能
SHOW	パラメータ値を表示する。
USE	メモリまたはディスクから、調査または変更のために一時作業領域に値を読み込む。
SET	パラメータ値を変更する。変更内容は作業領域のみで有効。変更内容をより永久的なものにするためには、WRITE コマンドが必要。
WRITE	作業領域の内容をメモリまたはディスクに書き込む。

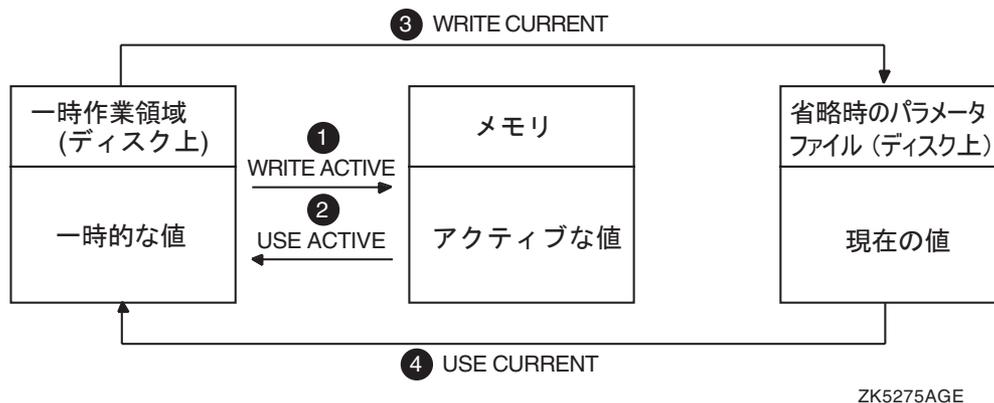
一時作業領域についての詳細は、1.8.1 項「パラメータ値と SYSGEN について」を参照してください。

1.8.1 パラメータ値と SYSGEN について

1.1.1 項「省略時の値、現在値、アクティブ値」で説明しているように、システム・パラメータの値にはいくつかの種類があります。簡単にまとめると、**現在値**とは、ディスク上の省略時のパラメータ・ファイルに格納されている値のことです。**アクティブ値**とは、メモリ内に格納され、システムの稼働中に使用される値のことです。これらの値のほかにも、SYSGEN はディスク上の独自の作業領域にパラメータ値を一時的に書き込みます。この値を一時値と呼びま

す。図 1-3 「SYSGEN パラメータの一時値, アクティブ値, 現在値」は, この 3 種類の値と, SYSGEN コマンドがそれらの値にどう影響するかを示しています。

図 1-3 SYSGEN パラメータの一時値, アクティブ値, 現在値



パラメータ値を表示または変更する場合, 通常のセッションでは次の手順に従います。

1. USE コマンドを使って, パラメータの値を SYSGEN の一時作業領域に読み込む。アクティブ値を読み込むときは USE ACTIVE, 現在値を読み込むときは USE CURRENT をそれぞれ使用する。
2. SHOW コマンドを使って, パラメータ値を表示する。
3. SET コマンドを使って, パラメータ値を変更する。ただし, SET コマンドは, SYSGEN の一時作業領域の値を変更するだけである。
4. WRITE コマンドを使って変更内容を有効にする。
 - WRITE ACTIVE は, メモリ内のアクティブ値のセットに値を書き込む。ただし, アクティブ値を変更できるのはダイナミック・パラメータの場合だけである。
 - WRITE CURRENT は, ディスク上の現在値のセットに値を書き込む。

システム・パラメータについての詳細は, 『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

1.8.2 SYSGEN によるパラメータ値の表示

システム・パラメータの値を表示するときには, 次の手順に従います。

1. 次のコマンドを入力して SYSGEN を起動する。

```
§ RUN SYS$SYSTEM:SYSGEN
```

2. 次の USE コマンドを使って, 表示する値を指定する。

表示対象	入力するコマンド
アクティブ値	USE ACTIVE
現在値	USE CURRENT
他のパラメータ・ファイルの値	USE ファイル指定

ファイル指定には, 表示したい値が含まれているパラメータ・ファイルの名前を指定する。たとえば, USE SYS\$SYSTEM:ALTPARAMS.DAT。

3. SHOW コマンドを次の形式で入力する。
SHOW [/修飾子] [パラメータ名]

特定のタイプのパラメータを表示する場合には、修飾子を指定する。例を示す。

表示対象となるグループ	入力するコマンド
WSMAX パラメータ	SHOW WSMAX
すべてのダイナミック・パラメータ	SHOW/DYNAMIC
TTY カテゴリのすべてのパラメータ	SHOW/TTY
すべてのパラメータ	SHOW/ALL

SYSGEN SHOW コマンドと修飾子についての詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』の SYSGEN の節を参照してください。

例

ここでは、SYSGEN を使ってすべての TTY システム・パラメータの現在値を表示します。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE CURRENT
SYSGEN> SHOW/TTY
```

```
Parameters in use: Current 1
Parameter Name          Current      Default      Min.         Max. Unit      Dynamic
-----
2          3          4          5          6          7
TTY_SCANDELTA          10000000    10000000    100000      -1 100Ns
TTY_DIALTYPE           0           0           0           255 Bit-Encode
TTY_SPEED              15          15          1           16 Special
TTY_RSPEED             0           0           0           16 Special
TTY_PARITY             24          24          0           255 Special
TTY_BUF               80          80          0           65535 Characters
TTY_DEFCHAR           402657952   402657952   0           -1 Bit-Encode
TTY_DEFCHAR2          135178      4098        0           -1 Bit-Encode
TTY_TYPAHDSZ          78          78          0           -1 Bytes
TTY_ALTYPAHD          2048        200         0           32767 Bytes
TTY_ALTALARM           750         64          0           -1 Bytes
TTY_DMASIZE           64          64          0           -1 Bytes D 8
TTY_CLASSNAME         "TTY"       "TTY"       "AA"        "ZZ" Ascii
TTY_SILOTIME          8           8           0           255 Ms
TTY_TIMEOUT           3600        900         0           -1 Seconds D
TTY_AUTOCHAR          7           7           0           255 Character D
SYSGEN>
```

SYSGEN は、次の情報を表示します。

- 1 使用中の値 (この例では、現在値)
- 2 システム・パラメータの名前
- 3 要求された値 (この例では、現在値)

このカラムの見出しは、パラメータの現在値を表示する場合にも、アクティブ値を表示する場合にも、常に "Current" である。この場合の "Current" は、USE コマンドで指定される、このパラメータの現在使用されている値を指すのであって、WRITE CURRENT コマンドによってディスクに格納されたパラメータの現在値を指すのではない。

- 4 省略時の値
- 5 最小値
- 6 最大値
- 7 割り当て単位
- 8 そのシステム・パラメータがダイナミックのときは "D"

1.8.3 SYSGEN によるシステム・パラメータ・ファイルの変更



重要:

システム生成ユーティリティ (SYSGEN) を使って変更されたパラメータ値は、AUTOGEN コマンド・プロシージャによって上書きされます。SYSGEN による変更内容を保持するには、SYS\$SYSTEM:MODPARAMS.DAT ファイルで新しいパラメータ値を指定する必要があります (1.5.1 項「MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御」を参照)。

注意:

システム・パラメータの変更は SYSGEN を使ってもできます。ただしシステム・パラメータ値を変更する場合にはできるだけ AUTOGEN を使ってください。詳細は 1.5 項「AUTOGEN によるシステム・パラメータの変更」を参照してください。

AUTOGEN を使用できない場合には、システム管理ユーティリティ (SYSMAN) を使用してください。詳細は 1.7 項「SYSMAN ユーティリティによるシステム・パラメータの管理」を参照してください。

省略時のシステム・パラメータ・ファイルの現在値の変更は、実行中のシステムのアクティブ値には、すぐには反映されません。次のシステム・ブート時に、新しい値で初期化されます。

例

VAX システム・パラメータ・ファイルの TTY_TIMEOUT パラメータの値を変更します。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN SYSGEN
SYSGEN> USE CURRENT
SYSGEN> SET TTY_TIMEOUT 3600
SYSGEN> WRITE CURRENT
%OPCOM, 15-APR-2000 16:04:06.30, message from user SYSTEM
%SYSGEN-I-WRITECUR, CURRENT system parameters modified by process
ID 00160030 into file VAXVMSSYS.PAR
SYSGEN> EXIT
```

1.8.4 SYSGEN によるアクティブ値の変更



重要:

SYSGEN によって変更されたパラメータ値は、AUTOGEN コマンド・プロシージャによって上書きされます。SYSGEN による変更内容を保持するためには、SYS\$SYSTEM:MODPARAMS.DAT ファイルで新しいパラメータ値を指定する必要があります (1.5.1 項「MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御」を参照)。

注意:

システム・パラメータの変更は SYSGEN を使ってもできます。しかし、システム・パラメータ値を変更する場合にはできるだけ AUTOGEN を使ってください。詳細は 1.7 項「SYSMAN ユーティリティによるシステム・パラメータの管理」を参照。

アクティブ値を変更すると、ダイナミック・パラメータにすぐに影響を与え、メモリ内にある値が変更されます。ダイナミック・パラメータについての詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル』を参照してください。また、SYSGEN の SHOW/DYNAMIC コマンドを使ってもダイナミック・パラメータであるか調べることができます。ダイナミック・パラメータ以外の値をシステムの稼働中に変更することはできません。アクティブ値を変更しても、ディスク上のシステム・パラメータの現在値には影響を与えません。次のシステム・ブート時に、それまでの現在値がアクティブ値として設定されます。

パラメータの新しいアクティブ値を設定し (WRITE ACTIVE と入力), 以降のブート時に新しい値を使用する場合には, 1.8.3 項 「SYSGEN によるシステム・パラメータ・ファイルの変更」で説明しているように, WRITE CURRENT コマンドを使用して, ディスク上の現在のパラメータ・ファイルに新しい値を書き込む必要があります。また, パラメータがダイナミック・パラメータでない場合には, WRITE CURRENT コマンドを入力して, システムをリブートする必要があります。

SYSGEN を使ってアクティブ・パラメータを変更すると, DCL の SET MESSAGE コマンドを使ってシステム・メッセージの形式を変更していない限り, オペレータ通信マネージャ (OPCOM) により, そのメッセージがオペレータ・ログとオペレータ・コンソールに書き込まれます。

例

1. PFCDEFAULT パラメータのアクティブ値を変更する。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN SYSGEN
SYSGEN> SET PFCDEFAULT 127
SYSGEN> WRITE ACTIVE
%OPCOM, 15-APR-2000 16:04:06.30, message from user SYSTEM
%SYSGEN-I-WRITEACT, ACTIVE system parameters modified by process
ID 00160030
SYSGEN> EXIT
```

2. PFCDEFAULT パラメータのアクティブ値を変更し, 同時に OpenVMS Alpha システム・パラメータ・ファイルに書き込む。このファイルは, システムのリブート時に使用される。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN SYSGEN
SYSGEN> SET PFCDEFAULT 127
SYSGEN> WRITE ACTIVE
%OPCOM, 15-APR-2000 16:04:06.30, message from user SYSTEM
%SYSGEN-I-WRITEACT, ACTIVE system parameters modified by process
ID 00160030
SYSGEN> WRITE CURRENT
%OPCOM, 15-APR-2000 16:04:06.30, message from user SYSTEM
%SYSGEN-I-WRITECUR, CURRENT system parameters modified by process
ID 00160030 into file ALPHAVMS$SYS.PAR
SYSGEN> EXIT
```

1.8.5 SYSGEN による新規パラメータ・ファイルの作成

パラメータ・ファイルを新しく作成しても, 稼働中のシステムには影響を与えません。しかし, それ以降の会話型ブート操作で, アクティブなシステムを新しいファイルの値で初期化することができます。

パラメータ・ファイルの作成方法

1. 次のコマンドを入力して SYSGEN を起動する。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN SYSGEN
```

2. 次の形式のコマンドを入力して, パラメータ・ファイルを SYSGEN の一時作業領域にコピーする。

USE ファイル指定

ファイル指定には, ベースとなる既存のパラメータ・ファイルを指定する。このファイルの値を変更して, 新しいパラメータ・ファイルを作成できる。

3. 次の形式のコマンドを入力して, 必要に応じて値を変更する。
SET パラメータ名 値

パラメータ名には、値を変更するパラメータの名前を指定する。値には、そのパラメータの新しい値を指定する。

4. 次の形式のコマンドを指定して、値を新しいパラメータ・ファイルに書き込む。
WRITE ファイル指定
ファイル指定には、作成するパラメータ・ファイルの名前を指定する。
5. SYSGEN を終了する。



重要:

SYSGEN ユーティリティを使って変更したパラメータ値は、AUTOGEN コマンド・プロシージャによって上書きされます。SYSGEN による変更内容を保持するためには、SYS\$SYSTEM:MODPARAMS.DAT ファイルに新しいパラメータ値を指定する必要があります (1.5.1 項「MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御」を参照)。

例

1. パラメータ・ファイル PARAMS.PAR の TTY_TIMEOUT パラメータの値を変更して、ファイルを更新する。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN SYSGEN
SYSGEN> USE SYS$MANAGER:PARAMS.PAR
SYSGEN> SET TTY_TIMEOUT 3600
SYSGEN> WRITE SYS$MANAGER:PARAMS.PAR
SYSGEN> EXIT
```

2. PARAMS.PAR ファイルをベースにして SYS\$SYSTEM:OURSITE.PAR という名前のファイルを作成する。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN SYSGEN
SYSGEN> USE SYS$MANAGER:PARAMS.PAR
SYSGEN> SET TTY_TIMEOUT 1000
SYSGEN> WRITE OURSITE.PAR
SYSGEN> EXIT
```

1.9 会話型ブートによるシステム・パラメータの変更



注意:

システム・パラメータの変更は会話型ブートで行うこともできます。しかし、システム・パラメータを変更する場合にはできるだけ AUTOGEN または SYSMAN ユーティリティを使用してください。詳細は、1.5 項「AUTOGEN によるシステム・パラメータの変更」および 1.7 項「SYSMAN ユーティリティによるシステム・パラメータの管理」を参照してください。

会話型ブートを使用するのは、他に影響を与えないシステム・パラメータを一時的に変更する場合か、緊急の場合だけに限定してください。たとえば、システムのアップグレード時に、会話型ブートを使用して、簡易スタートアップを使用するように STARTUP_P1 を変更します。

値を変更し、変更した値を AUTOGEN パラメータ・ファイル MODPARAMS.DAT に追加しない場合、AUTOGEN の次の実行時にその値は上書きされるということ覚えておいてください。

会話型ブート操作によって、システム・ブートの前にアクティブ・パラメータを変更できません。次の方法があります。

作業内容	参照箇所
個々のパラメータのアクティブ値を変更する。	『OpenVMS システム管理者マニュアル (上巻)』
省略時のパラメータ・ファイル以外のパラメータ・ファイルの値を使って、アクティブ値を初期化する。	『OpenVMS システム管理者マニュアル (上巻)』
省略時の値を使ってアクティブ値を初期化しなおす。	『OpenVMS システム管理者マニュアル (上巻)』

会話型ブートの最後に、省略時のパラメータ・ファイルが変更され、各パラメータの新しいアクティブ値が格納されます。



重要:

会話型ブートによって変更されたパラメータ値は、AUTOGEN コマンド・プロシージャによって上書きされます。会話型ブートによる変更内容を保持するためには、SYS\$SYSTEM:MODPARAMS.DAT ファイルに新しいパラメータ値を指定する必要があります (1.5.1 項「MODPARAMS.DAT による AUTOGEN パラメータ設定値の制御」を参照)。

1.10 BAP システム・パラメータのチューニング

OpenVMS Alpha バージョン 7.1 以降には、BAP (bus-addressable pool) の操作を制御するシステム・パラメータが含まれています。

Alpha システムでは、CIPCA, CIXCD, KFMSB, および Qlogic 1020ISP は、BAP (bus-addressable pool) を使用して性能を向上させるアダプタの一部です。BAP とは、I/O バスおよび 32 ビット・アダプタの物理アドレッシング・リミットを克服するために使用する、物理アドレス・チェック非ページング動的メモリ・プールです。

以下の表は、BAP の操作を制御するシステム・パラメータと、それらの省略時の値です。

システム・パラメータ	省略時の値
NPAG_BAP_MIN	0
NPAG_BAP_MAX	0

システム・パラメータ	省略時の値
NPAG_BAP_MIN_PA	0
NPAG_BAP_MAX_PA	-1

これらのパラメータの省略時の値を使用すると、システムは任意の構成でブートできます。構成されたシステムで AUTOGEN が実行されると、これらのパラメータは現在のシステム構成の性能を強化するような値に再設定されます。

インストール、アップグレード、または構成の変更の後でシステムがブートに失敗し、BAPパラメータが正しくないことを示すメッセージが表示された場合には、以下の手順を実行することをお勧めします。

1. BAP パラメータを省略時の値に再設定する。
2. システムをリブートする。
3. インストール手順で AUTOGEN を実行できるように設定するか、AUTOGEN を手作業で実行する。

AUTOGEN に FEEDBACK 修飾子を付けてこれらのパラメータを設定すると、次の例のようになります。

```
$ @SYS$UPDATE:AUTOGEN SAVPARAMS SETPARAMS FEEDBACK
```



注意:

これらのパラメータは重要です。ここで説明されているように AUTOGEN を実行し、これらのパラメータが正しく設定されているかどうかを確認することをお勧めします。

1つのBAPパラメータの設定だけを調整するので、このコマンドを使用しない場合には、次の手順を使用します。

1. 省略時の BAP パラメータ値を使用してシステムをブートする。
2. SYS\$SYSTEM:AGEN\$FEEDBACK.EXE を手作業で実行する。

```
$ RUN SYS$SYSTEM:AGEN$FEEDBACK.EXE
```

3. SYS\$SYSTEM:AGEN\$FEEDBACK.DAT で BAP_* システム・パラメータの値を検索する。

```
$ RUN SYS$SYSTEM:AGEN$FEEDBACK.EXE
```

4. Run SYSGEN を実行し、次のシステム・パラメータを、ステップ 3 で取得した BAP_ の値を使用して設定する。

AGEN\$FEEDBACK のデータ	システム・パラメータ	単位
BAP_MIN	NPAG_BAP_MIN	バイト
BAP_MAX	NPAG_BAP_MAX	バイト
BAP_MIN_PA	NPAG_BAP_MIN_PA	M バイト ¹
BAP_MAX_PA	NPAG_BAP_MAX_PA	M バイト ¹

¹ バージョン 7.2 以前の OpenVMS Alpha システムでは、このパラメータの値はバイト単位で指定されています。

BAP の割り当て量 (BAP_MIN および BAP_MAX によって指定) は、アダプタのタイプ、アダプタの数、およびオペレーティング・システムのバージョンによって異なります。物理アドレス範囲 (BAP_MIN_PA および BAP_MAX_PA によって指定) は、アダプタのタイプと、Galaxy 論理パーティションがあればそれが定義される方法によって異なります。



注意:

NPAG_BAP_MIN_PA および NPAG_BAP_MAX_PA を手作業で設定する場合には、必ずオペレーティング・システムのバージョンに対応する正しい単位 (バイトまたは M バイト) を使用して、各パラメータの値を設定してください。

第2章 ページ・ファイル，スワップ・ファイル，ダンプ・ファイルの管理

ページ・ファイル，スワップ・ファイル，ダンプ・ファイルは，自動的に作成されます。しかし，これらのファイルがどういうものを理解することは必要です。また，実際の環境に合わせて，これらのファイルを変更しなければならない場合があります。

この章の内容

この章では，次の作業について説明します。

作業	参照箇所
ページ・ファイルとスワップ・ファイルに関する情報の表示	2.3 項「ページ・ファイルとスワップ・ファイルに関する情報の表示」
各ファイルの適切なサイズの計算	2.4 項「ダンプ・ファイル，ページ・ファイル，スワップ・ファイルのサイズの机上計算」
ディスク空間が十分でない場合のダンプ・ファイル・サイズの最小化	2.5 項「ディスク空間が十分でない場合のシステム・ダンプ・ファイル・サイズの最小化」
複数バスまたはシャドウ・セット・メンバのあるシステム・ディスクでのダンプ・ファイルの書き込み	2.6 項「システム・ディスクへのシステム・ダンプ・ファイルの書き込み」
システム・ディスク以外へのデバイスへのダンプ・ファイルの書き込み	2.7 項「代替ディスクへのシステム・ダンプ・ファイルの書き込み」
クラッシュ・ダンプの内容を分析するための SDA の使用方法	2.8 項「SDA によるクラッシュ・ダンプの内容の分析」
クラッシュ・ダンプ要約情報の入手と分析のための SDA CLUE コマンドの使用法 ¹	2.9 項「SDA CLUE コマンドによるクラッシュ・ダンプ・ファイルの分析 (Alpha および I64)」
クラッシュ・ダンプについての履歴情報の入手のための CLUE の使用法 ²	2.10 項「CLUE を使用して，クラッシュ・ダンプに関する履歴情報を得る方法 (VAX のみ)」
システム障害発生後のシステム・ダンプ・ファイルの保存	2.11 項「システム障害後のシステム・ダンプ・ファイルの内容の保存」
テープまたはディスクへのダンプ・ファイルのコピー	2.12 項「システム・ダンプ・ファイルをテープまたはディスクへコピーする」
ページ・ファイルからのダンプ情報の解放	2.13 項「ページ・ファイルからのダンプ情報の解放」
ページ・ファイルとスワップ・ファイルのインストール	2.14 項「ページ・ファイルとスワップ・ファイルのインストール」
ページ・ファイル，スワップ・ファイル，ダンプ・ファイルの削除	2.15 項「ページ・ファイル，スワップ・ファイル，ダンプ・ファイルの削除」
ページ・ファイルとスワップ・ファイルの作成	2.16 項「ページ・ファイル，スワップ・ファイル，ダンプ・ファイルの作成と変更」
プロセス・ダンプへのアクセスの制御	2.17.2 項「プロセス・ダンプへのアクセスの許可 (Alpha および I64)」

- 1 Alpha および I64 のみ
- 2 VAX のみ

さらに、次の項目について説明します。

項目	参照箇所
ダンプ・ファイルについて	2.1 項「ダンプ・ファイルについて」
ページ・ファイルとスワップ・ファイルについて	2.2 項「ページ・ファイルおよびスワップ・ファイルについて」
選択型システム・ダンプ中の情報の順序について	2.5.1 項「選択型システム・ダンプでの情報の順序」
SDA CLUE について ¹	2.9.1 項「CLUE について (Alpha および I64)」
CLUE について ²	2.10.1 項「CLUE について (VAX のみ)」
プロセス・ダンプについて	2.17 項「プロセス・ダンプについて」

1 Alpha および I64 のみ

2 VAX のみ

2.1 ダンプ・ファイルについて

オペレーティング・システムは、修復不可能なエラー、あるいはシステム障害を発生させるような矛盾を内部で検出した場合、エラー・ログ・バッファ、プロセッサ、レジスタ、およびメモリの内容を**システム・ダンプ・ファイル**に書き込みます。システム・ダンプ・ファイルの以前の内容は、上書きされます。

Alpha システムでは、エラー・ログ・バッファの内容も、**エラー・ログ・ダンプ・ファイル**に書き込まれます。作成されたもののシステム・クラッシュ時には書き込まれていなかったエラー・ログ・エントリを含めて、リポート時にシステムを更新するために、エラー・ログ・ダンプ・ファイルは提供されます。

システム・ダンプ・ファイル

システム・ダンプ・ファイルを書き込む場合、システムは、いくつかのコンソール・メッセージとエラーまたは矛盾に関する情報を表示します。最後のメッセージは、ダンプ・ファイルが正常に書き込まれたことを示します。



重要:

コンソール・ターミナルを使用してシステムを停止する前に、終了メッセージが表示されるのを確認してください。これを行わないと、完全なシステム・ダンプ・ファイルを保存できない場合があります。

コンソール・メッセージとシステム・ダンプ・ファイルは、システム障害の原因を調べるための重要な情報源です。これらの情報は、次のように使用します。

- システム・ダンプ・アナライザ・ユーティリティ (SDA) を使ってダンプの内容を分析し、障害の原因を突き止める。
- Alpha システムおよび I64 システムでは、SDA CLUE コマンドを使って、ダンプ・ファイル要約情報を入手および分析する。
- VAX システムでは、CLUE を使用して、システム・ダンプ・ファイルから履歴情報を入手する。
- システム・ダンプの内容のコピーを作成し、弊社のサポート担当者にお知らせください。

オペレーティング・システムのディストリビューション・キットで提供されるシステム・ダンプ・ファイル `SYS$SPECIFIC:[SYSEX]SYSDUMP.DMP` はインストール中に作成されます。オペレーティング・システムは、システム・ダンプ・ファイルがなくても稼働します。しかし、シ

システムがクラッシュしたときは、クラッシュの原因を突き止めるために、ダンプ・ファイルが必要となります。AUTOGEN は、実際のハードウェア構成とシステム・パラメータに従って、システム・ダンプ・ファイルの適切なサイズを自動的に決定します。ディスク空間が十分でない場合のシステム・ダンプ・ファイル・サイズの最小化については、2.5 項「ディスク空間が十分でない場合のシステム・ダンプ・ファイル・サイズの最小化」を参照してください。

ハードウェア構成が特殊な場合、あるいは作業負荷が変化する場合には、システム・ダンプ・ファイルのサイズを変更することができます。詳細は 2.16.1 項「AUTOGEN を使用 (標準的な方法)」を参照してください。システム・ディスク以外のディスクに、システム・ダンプ・ファイルを書き込むことができます。これは、**Dump Off System Disk (DOSD)** と呼ばれます。詳細は 2.7 項「代替ディスクへのシステム・ダンプ・ファイルの書き込み」を参照してください。

エラー・ログ・ダンプ・ファイル

AUTOGEN は、インストール時にエラー・ログ・ダンプ・ファイルを作成します。ファイルのサイズは、システムの構成とシステム・パラメータによって異なります。VAX システム、Alpha システム、および I64 システムでのエラー・ログ・ダンプ・ファイルの違いは次のとおりです。

- Alpha システムおよび I64 システムでは、エラー・ログ・ダンプ・ファイルは `SYS$ERRLOG.DMP` と呼ばれます。このファイルはシステム・ディスクにあり、オペレータがシャットダウンを始めると、システムエラー・ログ・バッファの中身をエラー・ログ・ダンプ・ファイルに書き込みます。システム・ダンプ・ファイルには書き込みません。したがって、直前のシステム・クラッシュ・ダンプは上書きされません。
- VAX システムでは、エラー・ログ・ダンプ・ファイルは `SYSDUMP.DMP` と呼ばれます。システムがこのファイルをどのように扱うかは、システム管理者がダンプ・オフ・システム・ディスク (DOSD) を使用しているかどうかにかかわらずに依存します。
 - DOSD を使用している場合、エラー・ログはシステム・ディスクのスタブ・エラー・ログ・ダンプ・ファイル `SYSDUMP.DMP` に書き込まれます。また、エラー・ログとシステム・メモリは DOSD ディスクの `SYSDUMP.DMP` ファイルに書き込まれます。
 - DOSD を使用していない場合、エラー・ログとシステム・メモリはシステム・ディスクの `SYSDUMP.DMP` ファイルに書き込まれます。

VAX システムでは DOSD を使用しているかどうかにかかわらず、直前のシステム・クラッシュ・ダンプは、オペレータがシャットダウンを開始すると常に上書きされます。

2.1.1 ページ・ファイルを使ってシステム・クラッシュ・ダンプを格納する方法

オペレーティング・システムは、`SYS$SYSTEM:SYSDUMP.DMP` の最新バージョンにシステム・クラッシュ・ダンプを格納します。`SYSDUMP.DMP` が `SYS$SYSTEM` に存在した場合、オペレーティング・システムはシステム・ページング・ファイル `SYS$SYSTEM:PAGEFILE.SYS` の内容を書き換えます。

システム・パラメータ `SAVEDUMP` が設定されている場合、クラッシュ・ダンプ・ファイルはシステムのブート時、`PAGEFILE.SYS` に格納されます。`SAVEDUMP` がクリアされている場合、システムは、ページ・ファイルをページングに使用し、そのページング・ファイルに書き込まれていたダンプはなくなります。

`SYS$SYSTEM:PAGEFILE.SYS` ファイルにシステム・クラッシュ・ダンプを格納する場合には、その後で次のいずれかの方法を使って、システムのページングでダンプが占有している空間を使用できるように解放する必要があります。

- SDA の `COPY` コマンドを使用して、ダンプをページ・ファイルから別のファイルにコピーする。
- SDA の `RELEASE` コマンドを使用して、ページ・ファイルから情報を削除する。

詳細は 2.13 項「ページ・ファイルからのダンプ情報の解放」を参照してください。

適切なコマンドを SYSTARTUP_VMS.COM スタートアップ・コマンド・プロシージャに取り込んで、システムがリブートされるたびにページ・ファイルからダンプ情報を解放します。



重要:

ページ・ファイルを選択型ダンプに使用する場合は、注意が必要です。選択型ダンプでは、利用できるすべての空間が使用されるので、ページ・ファイルが小さい場合、選択型ダンプ情報でページ・ファイルがいっぱいになってしまい、システムのブート時にページング用の空間が残されていないこととなります。この結果、リブート時にシステムがハングすることがあります。

2.1.2 システム・ダンプの種類

システム・ダンプには、物理ダンプと選択型ダンプの 2 種類があります。表 2-1 「物理ダンプと選択型ダンプの比較」に、それらの比較を示します。また、表 2-3 「物理システム・ダンプ・ファイルと選択型システム・ダンプ・ファイルの比較」に、物理システム・ダンプ・ファイルと選択型システム・ダンプ・ファイルで利用できる情報の比較を示します。

表 2-1 物理ダンプと選択型ダンプの比較

ダンプの種類	説明
物理ダンプ	物理メモリのすべての内容をシステム・ダンプ・ファイルに書き込む。物理ダンプを確実に有効にするため、システム・ダンプ・ファイルには、物理メモリのすべての内容を含むのに十分なサイズが必要である。
選択型ダンプ	メモリのクラッシュ・ダンプの分析に役立つと考えられる部分を保存する。選択型システム・ダンプは、物理メモリ全体を保持できるだけのディスク空間がない場合に有効。

有効なシステム・ダンプを作成するために必要な条件

オペレーティング・システムにとって有効となるシステム・ダンプ・ファイルを保存するためには、次の条件が満たされている必要があります。

- システム・パラメータ DUMPBUG が 1 (省略時の値) に設定されている。
- システム・パラメータ SAVEDUMP が 0 (省略時の値) に設定されている場合、ファイル SYS\$SPECIFIC:[SYSEXE]SYSDUMP.DMP はシステム・ディスク上に存在しなければならない。
- ファイル SYS\$SPECIFIC:[SYSEXE]SYSDUMP.DMP がシステム・ディスク上に存在しない場合、ページ・ファイルを使用してダンプを格納しなければならない。システム・パラメータ SAVEDUMP は 1 に設定しなければならない。また、ファイル SYS\$SPECIFIC:[SYSEXE]PAGEFILE.SYS はシステム・ディスク上に存在しなければならない。
- ファイル SYS\$SPECIFIC:[SYSEXE]SYSDUMP.DMP が存在せず、SYS\$SPECIFIC:[SYSEXE]PAGEFILE.SYS をシステム・ダンプに使用できない場合には、ダンプ・ファイルを代替りのディスク上に作成しなければならない (2.7 項 「代替ディスクへのシステム・ダンプ・ファイルの書き込み」を参照)。
- メモリ全体を保持するための十分なディスク空間がない場合は、選択型ダンプを格納できるように、システム・パラメータ DUMPSTYLE を適切な値に設定しなければならない。詳細は 2.5 項 「ディスク空間が十分でない場合のシステム・ダンプ・ファイル・サイズの最小化」、および『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照。
- システム・ダンプ・ファイル (システム・パラメータ SAVEDUMP が設定されている場合にはページ・ファイル) が、システム障害の発生時に書き込まれるすべての情報を格納できるように十分大きくなければならない。

システム・パラメータ DUMPBUG が設定されていれば、ディスク空間が十分でない場合でも、AUTOGEN が SYSDUMP.DMP のサイズを自動的に変更する。

システム・パラメータ `SAVEDUMP` が設定されている場合、`AUTOGEN` がシステム・ダンプ・ファイルを操作することはない。

システム・パラメータ `SAVEDUMP` が設定されているかどうかにかかわらず、`AUTOGEN` がページ・ファイルのサイズを決定するときには、ページングしか考慮されない。

BACKUP 使用上の留意点

システム・ダンプ・ファイルには、`NOBACKUP` 属性が設定されています。したがって、ダンプ・ファイルをコピーする場合には、`BACKUP` の起動時に `/IGNORE=NOBACKUP` 修飾子を使用する必要があります。SDA の `COPY` コマンドを使用してシステム・ダンプ・ファイルをコピーする場合、コピー先のファイルに `NOBACKUP` 属性は設定されません。コピーに `NOBACKUP` 属性を設定したい場合には、`SET FILE` コマンドに `/NOBACKUP` 修飾子を指定します (詳細は『OpenVMS DCL ディクショナリ』を参照してください)。

機密保護の問題点

省略時の設定では `SYS$SYSTEM:SYSDUMP.DMP` は、一般ユーザがアクセスできないようになっています。システム・ダンプ・ファイルには特権情報が入っている可能性があるため、システム・ダンプ・ファイルのこの保護レベルはそのままにしておいてください。同様に、2.11 項「システム障害後のシステム・ダンプ・ファイルの内容の保存」および 2.13 項「ページ・ファイルからのダンプ情報の解放」で説明するように、システム・ダンプ・アナライザ・ユーティリティ (SDA) を使って、システム・ダンプ・ファイルをコピーする場合は、このコピーに必ず保護を設定して、一般ユーザがアクセスできないようにしてください。ファイル保護についての詳細は、『OpenVMS システム・セキュリティ・ガイド』を参照してください。

2.2 ページ・ファイルおよびスワップ・ファイルについて

メモリを有効に使用するため、オペレーティング・システムは、物理メモリからディスクへ、またはディスクから物理メモリへ、情報を移動します。このようなメモリ管理は 2 種類あり、ページングとスワッピングと呼ばれます。表 2-2 「ページングとスワッピングに関連する用語」に、ページングとスワッピングに関連する用語を示します。

表 2-2 ページングとスワッピングに関連する用語

用語	定義
ページング	プロセスに割り当てられた物理メモリを効率よく利用するためのメモリ管理操作。ページングによって、プロセス作業領域の使用頻度の低い部分が、物理メモリからファイルに移動する。ページングについての詳細は、『Guide to OpenVMS Performance Management』を参照。
ページ・ファイル	ページングされたメモリの部分を書き込まれるファイル。OpenVMS のインストール・プロセスによって、 <code>SYS\$SYSTEM:PAGEFILE.SYS</code> という名前のページ・ファイルができる。必要であれば、 <code>SYS\$SYSTEM:PAGEFILE.SYS</code> をシステム・クラッシュ・ダンプ・ファイルの代わりに使用できる。詳細は 2.1.1 項「ページ・ファイルを使ってシステム・クラッシュ・ダンプを格納する方法」を参照。
スワップ	システム全体で使用できる物理メモリを効率よく利用するためのメモリ管理機能。スワップによって、活動頻度の低いプロセスの作業領域全体が、物理メモリからファイルに移動する。スワッピングについての詳細は、『Guide to OpenVMS Performance Management』を参照。
スワップ・ファイル	スワッピングされたメモリの部分を書き込まれるファイル。OpenVMS のインストール・プロセスによって、 <code>SYS\$SYSTEM:SWAPFILE.SYS</code> という名前のスワップ・ファイルができる。

表 2-2 ページングとスワッピングに関連する用語 (続き)

用語	定義
1 次ページ・ファイルと 1 次スワップ・ファイル	OpenVMS のインストール中にできる省略時のページ・ファイル (SYS\$SYSTEM:PAGEFILE.SYS) とスワップ・ファイル (SYS\$SYSTEM:SWAPFILE.SYS)。
2 次ページ・ファイルと 2 次スワップ・ファイル	性能とディスク空間の理由により、ユーザによって作成される付加的なページ・ファイルとスワップ・ファイル。1 次ページ・ファイルと 1 次スワップ・ファイルをシステム・ディスク上に保持している場合、システムは、1 次ページ・ファイルと 1 次スワップ・ファイルの空間に加え、ページングとスワッピングのための 2 次ファイルの空間を使用する。2 次ページ・ファイルと 2 次スワップ・ファイルの作成法については、2.16 項「ページ・ファイル、スワップ・ファイル、ダンプ・ファイルの作成と変更」を参照。

ファイルのインストール

ページ・ファイルとスワップ・ファイルをインストールしてからでないと、システムはそれらのファイルを使用することはできません。スタートアップ時にシステムは自動的に、SYS\$SYSTEM:PAGEFILE.SYS と SWAPFILE.SYS の最新バージョンをインストールします。2 次ページ・ファイルと 2 次スワップ・ファイルを作成する場合は、スタートアップ時にシステムがそれらを確実にインストールするようにしなければなりません。ページ・ファイルとスワップ・ファイルのインストールについての詳細は、2.14 項「ページ・ファイルとスワップ・ファイルのインストール」を参照してください。

ファイルのサイズと記憶位置

AUTOGEN は、実際のハードウェア構成とシステム・パラメータに従い、これらのファイルの適切なサイズを決定します。しかし、特殊な構成のシステム、あるいは作業負荷の変化が大きいシステムでは、ページ・ファイルまたはスワップ・ファイルのサイズを変更しなければならないことがあります。詳細は 2.16.1 項「AUTOGEN を使用 (標準的な方法)」を参照してください。

システムにシステム・クラッシュ・ダンプを格納するためのページ・ファイルが必要ない場合は、システム・ディスクからページ・ファイルを削除することができます。ただし、できればシステム・ディスクにページ・ファイルを 1 つ残しておいて、ページ・ファイルを保持している別のディスクが使用できない場合に、システムをブートできるようにしておいた方がよいでしょう。また、スワップ・ファイルもシステム・ディスクから削除することができます。

2.3 ページ・ファイルとスワップ・ファイルに関する情報の表示

DCL の SHOW MEMORY/FILES コマンドにより、システムに存在しているページ・ファイルとスワップ・ファイルに関して、ファイル名、サイズ、使用している空間の量などの情報が表示できます。次に例を示します。

```

$ SHOW MEMORY/FILES
      System Memory Resources on 19-JAN-2001 13:35:26.58

Swap File Usage (8KB pages):
  DISK$PAGE_DUMPS: [SYS0.SYSEX] SWAPFILE.SYS;2
                                     Index      Free      Size
                                     1          7992     8248

Paging File Usage (8KB pages):
  DISK$PAGE_DUMPS: [SYS0.SYSEX] PAGEFILE.SYS;1
                                     Index      Free      Size
                                     254       13722    16496

Total committed paging file usage:
                                     4870
  
```

コミットされたページングファイルの合計使用量は、ページングに必要なページ・ファイル領域を必要とするシステムでのページ数です。この値は、使用可能なページファイルの合計ページ数より大きくなって構いません。すべての必要な領域がページングで一度に使用されるようなことが起こる可能性は低いからです。

2.4 ダンプ・ファイル，ページ・ファイル，スワップ・ファイルのサイズの机上計算

ページ・ファイル，スワップ・ファイル，ダンプ・ファイルのサイズは，オペレーティング・システムのインストールまたはアップグレード時に，AUTOGENによって自動的に計算されます。しかし，必要に応じて，これらのファイルのサイズを机上で計算することもできます。次に，ページ・ファイル，スワップ・ファイル，ダンプ・ファイルの適切なサイズを決定する方法を説明します。

2.4.1 システム・ダンプ・ファイルのサイズの計算

完全なクラッシュ・ダンプを保存するために，システム・ダンプ・ファイルのサイズが十分であることを確認してください。ダンプ・ファイルの適切なサイズは，AUTOGEN コマンド・プロシージャにより計算されます。ただし，ダンプ・ファイル・サイズを机上で計算したい場合は，次の公式を使用してください。この公式により，物理ダンプに保持する必要のあるサイズが計算できます。

SYSDUMP.DMP の場合

VAX システムの場合には，次の公式を使用します。

ファイルのブロック数 (SYS\$SYSTEM:SYSDUMP.DMP)
= 物理メモリのページ数
+ (エラー・ログ・バッファ数 * バッファあたりのブロック数)
+ 1

Alpha システムおよび I64 システムの場合には，次の公式を使用します。

ファイルのブロック数 (SYS\$SYSTEM:SYSDUMP.DMP)
= 物理メモリのページ数 * ページあたりのブロック数
+ (エラー・ログ・バッファ数 * バッファあたりのブロック数)
+ 10

ページ数 物理メモリのサイズ (ページ単位)。システムの物理メモリ全体のサイズを決めるには DCL の SHOW MEMORY コマンドを使用する。

ページあたりのブロック数 メモリのページあたりのブロック数。
Alpha システムおよび I64 システムの場合，メモリの各ページのブロック数は，システムのページ・サイズを 512 (ブロック・サイズ) で割ることによって求める。次のコマンドを使用する。

```
$ PAGESIZE==F$GETSYI ("PAGE_SIZE")  
$ BLOCKSPERPAGE=PAGESIZE/512  
$ SHOW SYMBOL BLOCKSPERPAGE
```

エラー・ログ・バッファの数 システム・パラメータ ERRORLOGBUFFERS の値。このパラメータにより，エラー・ログ・バッファの数が設定され，メモリ内で永久的に割り当てられる。

バッファあたりのブロック数 システム・パラメータ ERLBUFFERPAGES の値。このパラメータにより，各バッファ内のメモリのページレット (ブロック) 数が設定される。

メモリ・サイズが大きなシステム，あるいはディスク容量の小さなシステムでは，完全メモリ・ダンプを行うのに十分な空間がとれないことがあります。そのような環境では，特定の情報だけをダンプするように，システム・パラメータ DUMPSTYLE の値を設定します。詳細は 2.5 項「ディスク空間が十分でない場合のシステム・ダンプ・ファイル・サイズの最小化」を参照してください。

PAGEFILE.SYS の場合

SYS\$SYSTEM:SYSDUMP.DMP がない場合，クラッシュ・ダンプは，1 次ページ・ファイル SYS\$SYSTEM:PAGEFILE.SYS に書き込まれます。ページ・ファイルの適切なサイズは，AUTOGEN

コマンド・プロシージャにより計算されます。ただし、クラッシュ・ダンプを保持するために必要なページ・ファイルの最小サイズを机上で計算したい場合は、次の公式を使用します。

VAX システムの場合には、次の公式を使用します。

ファイルのブロック数 (SYS\$SYSTEM:PAGEFILE.SYS)
= 物理メモリのページ数
+ (エラー・ログ・バッファ数 * バッファあたりのブロック数)
+ 1
+ 1000

Alpha システムおよび I64 システムの場合には、次の公式を使用します。

ファイルのブロック数 (SYS\$SYSTEM:PAGEFILE.SYS)
= 物理メモリのページ数 * ページあたりのブロック数
+ (エラー・ログ・バッファ数 * バッファあたりのブロック数)
+ 物理メモリのページ数 / 512
+ 10
+ システム・パラメータ RSRVPAGCNT の値

ページ数	物理メモリのサイズ (ページ単位)。システムの物理メモリ全体のサイズを決めるには DCL の SHOW MEMORY コマンドを使用する。
ページあたりのブロック数	メモリのページあたりのブロック数。 Alpha システムおよび I64 システムの場合、メモリの各ページのブロック数は、システムのページ・サイズを 512 (ブロック・サイズ) で割ることによって求める。次のコマンドを使用する。 <pre>\$ PAGESIZE==F\$GETSYI ("PAGE_SIZE") \$ BLOCKSPERPAGE=PAGESIZE/512 \$ SHOW SYMBOL BLOCKSPERPAGE</pre>
エラー・ログ・バッファの数	システム・パラメータ ERRORLOGBUFFERS の値。このパラメータにより、エラー・ログ・バッファの数が設定され、メモリ内で永久的に割り当てられる。
バッファあたりのブロック数	システム・パラメータ ERLBUFFERPAGES の値。このパラメータにより、各バッファ内のメモリのページレット (ブロック) 数が設定される。
RSRVPAGCNT	RSRVPAGCNT 特殊システム・パラメータの値。



重要:

この公式では、ダンプを保存するための 1 次ページ・ファイルの最低限のサイズだけが求められます。ほとんどのシステムでは、システムがハングしないようにページ・ファイルのサイズをこの値より大きくする必要があります。ページ・ファイル・サイズの計算についての詳細は、2.4.3 項「ページ・ファイルのサイズの計算」を参照してください。

2.4.2 エラー・ログ・ダンプ・ファイルのサイズの計算

これらの計算は、OpenVMS VAX システム、Alpha システム、および I64 システムでは異なります。

Alpha システムおよび I64 システム

Alpha システムおよび I64 システムでは、AUTOGEN コマンド・プロシージャはエラー・ログ・ダンプ・ファイルの適切なサイズを計算します。ただし、エラー・ログ・ダンプ・ファイルのサイズを机上で計算したい場合は、次の公式を使用してください。この公式により、すべてのエラー・ログ・バッファを保持するために必要なサイズが計算できます。

ファイルのブロック数 (SYS\$SYSTEM:SYS\$ERRLOG.DMP)
= エラー・ログ・バッファ数 * バッファあたりのブロック数

+ 2

- | | |
|----------------------|---|
| エラー・ログ・バッファの数 | システム・パラメータ ERRORLOGBUFFERS の値。このパラメータは、メモリで永久的に割り当てられるエラー・ログ・バッファの数を設定する。 |
| バッファあたりのブロック数 | システム・パラメータ ERLBUFFERPAGES の値。このパラメータは、各バッファ内のメモリのページレット (ブロック) 数を設定する。 |

VAX システム

VAX システムでは、エラー・ログ・ダンプ・ファイルのサイズはダンプ・オフ・システム・ディスク (DOSD) を使用しているかどうかで異なります。

- DOSD を使用していない場合: エラー・ログはシステム・ディスクのダンプ・ファイル SYSDUMP.DMP に書き込まれます。サイズ情報については、2.4.1 項「システム・ダンプ・ファイルのサイズの計算」を参照してください。
- DOSD を使用している場合:
 - エラー・ログとシステム・メモリの両方が DOSD ディスク上の SYSDUMP.DMP ファイルに書き込まれます。サイズ情報については、2.4.1 項「システム・ダンプ・ファイルのサイズの計算」を参照してください。
 - さらに、エラー・ログはシステム・ディスクのスタブ・エラー・ログ・ダンプ・ファイル SYSDUMP.DMP に書き込まれます。これは 2048 ブロックの固定長ファイルであるため、サイズ計算が必要ではありません。

2.4.3 ページ・ファイルのサイズの計算

システム性能を維持するためには、ページ・ファイルの空間が十分にあることが重要です。ページ・ファイル空間の適切なサイズは、AUTOGEN コマンド・プロシージャにより計算されます。AUTOGEN は十分なサイズを算出するはずですが、ページ・ファイル空間のサイズを机上で計算したい場合には、次のいずれかの公式を使用します。

VAX システム

VAX システムの場合には、次の公式を使用して、ページ・ファイル空間のサイズを算出します。

ブロックのサイズ
(システム上のすべてのページ・ファイルの合計)
= サイトの平均プロセス・サイズ (ページ数)
* プロセスの最大数

- **平均プロセス・サイズ**は、プロセスの平均仮想サイズの値。平均プロセス・サイズを求めるには、次のコマンドを使用する。

```
$ SHOW PROCESS/CONTINUOUS/ID=pid
```

値は、ページ単位で指定する。

- **プロセスの最大数**は MAXPROCESSCNT システム・パラメータの値。

調整の結果が VIRTUALPAGECNT より少ない場合には、代わりに VIRTUALPAGECNT の値を使用します。

システムの仮想ページの数を調べるには、次のコマンドを入力します。

```
$ WRITE SYS$OUTPUT F$GETSYI ("VIRTUALPAGECNT")
```

Alpha システムおよび I64 システム

Alpha システムおよび I64 システムの場合には、メモリ・サイズと用途にさまざまなバリエーションがあるので、単純な公式はありません。512 MB までのシステムでは、次の公式を使用してページ・ファイル・サイズを算出します。

ブロック・サイズ
(システムのすべてのページ・ファイルの合計)
= 平均プロセス・サイズ (ページ単位)
* ページ単位のブロック数
* 最大プロセス数

- **平均プロセス・サイズ**は、プロセスの平均仮想サイズの値。平均プロセス・サイズを求めるには、次のコマンドを使用する。

```
$ SHOW PROCESS/CONTINUOUS/ID=pid
```

値は、ページ単位で指定する。

- システム・ページ・サイズを 512 (ページレット・サイズ) で割ることによって、**ページ単位のブロック数**を計算する。たとえば、ページ・サイズ 8192 のシステムは、ページ単位で 16 ページレットを持つことになる。

システムのページ・サイズを決めるには、次のコマンドを入力する。

```
$ WRITE SYS$OUTPUT F$GETSYI ("PAGE_SIZE")
```

- **プロセス最大数**は、システム・パラメータ MAXPROCESSCNT の値になる。

512MB を超えるシステムでは、以下の項で説明するページ・ファイル使用状況の監視の手順 (2.4.3.2 項「ページ・ファイル使用状況の監視」) に従って、必要に応じて調整してください。

2.4.3.1 ページ・ファイル・サイズの表し方

算出したページ・ファイルは、次のいずれかの方法で表すことができます。

- 1 次ページ・ファイルのみ
- 1 次ページ・ファイルと 2 次ページ・ファイルに振り分ける。
- SYS\$SYSTEM 内の 1 次ページ・ファイルを削除した場合は、複数の 2 次ページ・ファイル間で振り分ける。

2.4.3.2 ページ・ファイル使用状況の監視

(AUTOGEN を使って、あるいは机上の計算で) ページ・ファイルの初期のサイズが決まったら、次のコマンドで AUTOGEN を実行してページ・ファイルの使用状況を監視してください。

```
$ @SYS$UPDATE:AUTOGEN SAVPARAMS TESTFILES FEEDBACK
```

このコマンドを実行すると、AUTOGEN によりページ・ファイルの使用量とサイズの推奨値がフィードバック・レポート AGEN\$PARAMS.REPORT に書き込まれます。なお、AUTOGEN およびフィードバック・レポートの詳細については、1.4 項「AUTOGEN コマンド・プロシージャについて」と 1.4.2 項「フィードバック・レポート (AGEN\$PARAMS.REPORT) について」を参照してください。2.3 項「ページ・ファイルとスワップ・ファイルに関する情報の表示」で説明したように、また、DCL の SHOW MEMORY/FILES コマンドを使用すると、ファイルの使用状況も表示されます。

ページ・ファイルの使用量が 1 次ページ・ファイルのサイズ (または 1 次ページ・ファイルと 2 次ページ・ファイルのサイズを合計したサイズ) の半分以上にならないようにします。ページ・ファイルの使用量がシステム性能に影響するレベルに近づくと、コンソール・ターミナルにメッセージが出力されます。その場合は、ページ・ファイルのサイズを大きくするか、ファイルを追加してください。



注意:

システムの資源と代表的な作業負荷は、ページ・ファイルの必要なサイズに影響します。そのため、これらの要素についてよく理解しておく必要があります。詳細は『Guide to OpenVMS Performance Management』を参照してください。

2.4.3.3 ページ・ファイル空間の制限

AUTHORIZE の ADD および MODIFY コマンドに /PGFLQUOTA 修飾子を指定して、ユーザ・プログラムが使用するページ・ファイルの量を制限してください (詳細は『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』の AUTHORIZE の節を参照)。
/PGFLQUOTA の値は、必ず 1024 以上にしてください。ページ・ファイル空間の必要量は、ユーザのアプリケーションによって大きく異なります。

2.4.4 スワップ・ファイル・サイズの計算

システム性能を維持するためには、スワップ・ファイルの空間が十分にあることが重要です。スワップ・ファイルの空間に適したサイズは、AUTOGEN コマンド・プロシージャにより計算されます。スワップ・ファイルの空間のサイズを机上で計算したい場合は、次の公式を使用してください。

ファイルのサイズ

(システム上のすべてのスワップ・ファイルの合計)

= プロセスの最大数 (システム・パラメータ MAXPROCESSCNT)

* システム上のプロセスの平均ワーキング・セット・クォータ

プロセスの最大数

MAXPROCESSCNT システム・パラメータの値。

**システム上のプロセスの平均
ワーキング・セット・クォ
ータ**

システム上で動作しているプロセスの WSQUOTA 制限の平均値。

VAX システムの場合は、ページ単位で指定する。

Alpha システムおよび I64 システムの場合は、ページレット単位で指定する。

2.4.4.1 スワップ・ファイル・サイズの表し方

Alpha システムと VAX システムでは、算出したサイズを次のような方法で表すことができません。

- 1 次スワップ・ファイルのみ
- 1 次スワップ・ファイルと 2 次スワップ・ファイルに振り分ける。
- SYS\$SYSTEM 内の 1 次スワップ・ファイルを削除した場合は、複数の 2 次スワップ・ファイル間で振り分ける。

2.4.4.2 スワップ・ファイルの使用状況の監視

(AUTOGEN を使って、あるいは机上計算によって) スワップ・ファイルの空間の適切なサイズが決まったら、2.3 項「ページ・ファイルとスワップ・ファイルに関する情報の表示」で説明したように、DCL の SHOW MEMORY/FILES コマンドを使って、スワップ・ファイルの使用状況を監視してください。スワップ・ファイル空間の 1/3 は未使用のままにしてください。そうしないと、システム性能が著しく低下します。



注意:

システムの資源と作業負荷は、ページ・ファイルの必要なサイズに影響します。そのため、これらの要素についてよく理解しておく必要があります。詳細は『Guide to OpenVMS Performance Management』を参照してください。

2.5 ディスク空間が十分でない場合のシステム・ダンプ・ファイル・サイズの最小化

システム構成によっては、ディスク・ファイルにメモリ全体の内容を保存できないことがあります。たとえば、メモリ・サイズが大きなシステムでは、完全メモリ・ダンプに十分なディスク空間が確保できないことがあります。このような場合には、システム・ダンプ・アナライザ (SDA) でダンプを分析できません。

また、VAXシステムの場合、ダンプ空間の不足が原因で、クラッシュ・ログ・ユーティリティ・エキストラクタ (CLUE) もダンプを分析できなくなります。

システム・ダンプ・ファイルのサイズ最小化のためのオプション

ディスク容量が不足しているときに、システム・ダンプ・ファイルのサイズを最小化するには、次に示すオプションのいずれか1つを使用します。

- 選択型ダンプ

VAXシステムと Alpha システムでは、システム障害の原因を突き止めるために最も有用な情報を含む部分のメモリを保持するために、選択型システム・ダンプを使用することができます。物理ダンプと選択型ダンプの比較を表 2-1 「物理ダンプと選択型ダンプの比較」に示します。

また、表 2-3 「物理システム・ダンプ・ファイルと選択型システム・ダンプ・ファイルの比較」には、物理ダンプ・ファイルと選択型システム・ダンプ・ファイルで使用できる情報の比較を示します。

表 2-3 物理システム・ダンプ・ファイルと選択型システム・ダンプ・ファイルの比較

ダンプの種類	入手できる情報	入手できない情報
物理ダンプ(またはフル・ダンプ)	物理アドレスとエラー・ログ・バッファの順序で格納された、使用中のメモリ全体の内容。	クラッシュ時、メモリからディスクにページングされていた内容。
選択型ダンプ	システム・ページ・テーブル、システム空間メモリ、エラー・ログ・バッファ、保存されているすべてのプロセスのプロセス領域と制御領域 (および グローバル・ページ)。	クラッシュ時にメモリからディスクにページングされていた内容、保存されていないプロセスのプロセス領域と制御領域、ページ・テーブルによってマップされていないメモリ。

選択型システム・ダンプを保存するには、システム・パラメータ DUMPSTYLE のビット 0 を適切な値に設定します。システム・パラメータとその値については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の付録に記載されています。システム・パラメータ値の変更については、1.5 項「AUTOGEN によるシステム・パラメータの変更」を参照してください。

- 圧縮ダンプ

Alpha システムおよび I64 システムで、DUMPSTYLE システム・パラメータのビット 3 をセットすると、OpenVMS は物理システム・ダンプまたは選択型システム・ダンプを圧縮形式で書き込みます (圧縮の正確な量はシステムの使用状況に応じて異なりますが、典型的な圧縮ダンプは元のサイズの約 60 パーセントです)。圧縮ダンプを使用する場合には、AUTOGEN はシステム・ダンプ・ファイルのサイズを、圧縮しない場合のサイズの 2/3 に設定します。

- ダンプ・オフ・システム・ディスク (DOSD)
DUMPSTYLE システム・パラメータのビット 2 をセットし、他のすべての必要条件を満たすと、OpenVMS はシステム・ダンプをシステム・ディスク以外のディスクに書き込みます。詳細については、2.7 項「代替ディスクへのシステム・ダンプ・ファイルの書き込み」の説明を参照してください。

2.5.1 項「選択型システム・ダンプでの情報の順序」では、Alpha システム、I64 システム、および VAX システムで選択型システム・ダンプに情報が書き込まれる順序について説明しています。2.5.2 項「選択型システム・ダンプにプロセスが書き込まれる順序の微調整 (Alpha および I64)」では、Alpha システムおよび I64 システムでこの順序を細かく調整する方法について説明しています。

2.5.1 選択型システム・ダンプでの情報の順序

VAX システム、Alpha システム、および I64 システムで選択型ダンプに情報が書き込まれる順序は次のとおりです。

VAX システムでは、情報は次の順序で選択型ダンプに書き込まれます。

1. システム・ページ・テーブル (SPT)
2. システム空間(プロセス・ページ・テーブル、ページ・フレーム番号 (PFN) データベース、およびグローバル・ページ・テーブル (GPT) を含む)
3. プロセスのワーキング・セット内のグローバル・ページ
4. クラッシュが発生した時点で常駐していたプロセス
 - a. クラッシュした CPU の現在のプロセス
 - b. あらかじめ定義されたプロセス (BUGCHECK にハード・コーディングされる)
 - c. 他の CPU での現在のプロセス
 - d. クラッシュが発生した時点で常駐していた他のプロセス (プロセス・インデックスの順)

Alpha システムおよび I64 システムでは、情報は次の順序で選択型システム・ダンプに書き込まれます。

1. 共有アドレス (S0/S1/S2) のページ・テーブル (PT)
2. S0/S1 空間
3. S2 空間
4. 複製されたページの内容が元のページと異なっている、性能上の理由から複製されているシステム空間ページ (P1, S0/S1, S2)
5. 適切な場合には、Galaxy 共有メモリ領域のメモリ・マップ・ページ
6. キー・プロセス
 - a. クラッシュした CPU の現在のプロセス
 - b. スワップ
 - c. CPU 上の現在のプロセスのうち、クラッシュ状態を記録できなかったもの
 - d. 他の CPU の現在のプロセス
 - e. 使用しているシステム固有の優先プロセス (次の節を参照)
 - f. 弊社が定義している優先プロセス (BUGCHECK にハード・コーディングされる)
 - MSCPmount
 - AUDIT_SERVER
 - NETACP
 - NET\$ACP
 - REMACP
 - LES\$ACP
7. リソースまたはその他の待ち状態での任意のプロセス (RWAST など)

8. キー・グローバル・ページ (キー・プロセスのワーキング・セット内のグローバル・ページ)
9. クラッシュが発生した時点で常駐していた他のプロセス (非キー・プロセス)。ただしプロセス・インデックスの順。
10. 非キー・プロセスのワーキング・セット内の残りのグローバル・ページ

Alpha システムおよび I64 システムでは、プロセスは 2 段階でダンプされます。最初にプロセスのページ・テーブルがダンプされ、次にプロセスのボディがダンプされます。

Alpha システム, I64 システム, および VAX システムでの使用法に関する注意

Alpha プラットフォーム, I64 プラットフォーム, VAX プラットフォームのいずれでも、プロセスが 2 回ダンプされることはありません。たとえば、Alpha システムおよび I64 システムでは、現在のプロセスがスワップの場合、それは 1 回だけダンプされます。

同様に、Alpha システムおよび I64 システムでグローバル・ページが 2 回ダンプされることはありません。したがって、キー・プロセスのワーキング・セット内のページが“キー・グローバル・ページ”セクションでダンプされた場合には、それが非キー・プロセスのワーキング・セットにも存在するからといって、後でもう一度ダンプされることはありません。

2.5.2 選択型システム・ダンプにプロセスが書き込まれる順序の微調整 (Alpha および I64)

Alpha システムおよび I64 システムでは、**キー・プロセス**と呼ばれる一連のプロセスは、そのプロセスに逆リンクする遷移ページも含めて、PT, S0/S1, S2 の直後にダンプされます。システム管理者は、キー・プロセスとして取り扱う追加プロセスを指定できます。これらのプロセスには、ダンプで他のプロセスより高い優先順位が与えられます。したがって、ダンプ・ファイルが小さすぎて、すべてのプロセスを格納できない場合でも、選択したプロセスは正しく書き込まれます。

作業方法

ダンプするプロセスの順序を指定するには、SYSMAN DUMP_PRIORITY コマンドを使用します。

- DUMP_PRIORITY ADD - ダンプの初期段階でダンプされるプロセスの一覧に、プロセスを追加します。
- DUMP_PRIORITY LOAD - リストのイン・メモリ・コピーをアップデートします。
DUMP_PRIORITY LOAD コマンドは、システムのスタートアップ中に自動的に起動されます。

システムが稼働している間であればいつでも、新しいプロセスの追加や、リストのイン・メモリ・コピーのアップデートを行うことができます。したがって、プロセスがハングした場合には、システム管理者はプロセスを優先プロセスとして指定し、強制的にクラッシュを発生させることができます。

2.6 システム・ディスクへのシステム・ダンプ・ファイルの書き込み

システム・ディスクへのパスが 2 つ以上あるか、システム・ディスクが複数のメンバを持つシャドウ・セットである場合には、確実にシステム・ダンプをシステム・ディスクに書き込むことができるようにするための追加策を取らなければなりません。

2.6.1 Alpha システムおよび I64 システムでのシステム・ディスクへのシステム・ダンプ

システム・ディスクへのパスが 2 つ以上ある場合には、コンソール環境変数 DUMP_DEV に、そのシステム・ディスクへのすべてのパスが記述されていなければなりません。このようにすれば、フェールオーバーのため元のブートパスが使用不可能になった場合でも、システムは引き続きシステム・ディスクを検索し、そこにシステム・ダンプを書き込むことができます。

システム・ディスクのシャドウ・セットが複数のメンバを持つ場合には、コンソール環境変数 DUMP_DEV に、そのシャドウ・セットのすべてのメンバへのすべてのパスが記述されていなければなりません。このようにすれば、マスタ・メンバが代わった場合でも、システムは引き続きマスタ・メンバを検索し、そこにシステム・ダンプを書き込むことができます。

DUMP_DEV を定義しない場合には、システムは、ブート時に使用されたものと同じ物理バスだけを使用した場合に限り、ブート時に使用された物理ディスクにのみ、システム・ダンプを書き込むことができます。DUMP_DEV の設定の詳細については、2.7.1 項「Alpha システムおよび I64 システムでの DOSD の必要条件」を参照してください。

システム・ダンプ・ファイルを代替ディスクへ書き込むこともできますが (2.7.1 項「Alpha システムおよび I64 システムでの DOSD の必要条件」を参照)、その場合にも、エラー・ログを書き込むために、システム・ディスクへのパスを定義しなければなりません。したがって、DUMP_DEV には、代替ダンプ・ディスクへのパスの他に、システム・ディスクへのすべてのパスも含まれている必要があります。

Alpha システムでは、一部の構成 (FC (Fibre Channel) ディスクを使用している場合など) には、システム・ディスクへのパスの組み合わせが、DUMP_DEV にリストできる以上に含まれている場合があります。そのような場合には、通常はシャドウ・セットのマスタ・メンバであるシステム・ディスクへのすべてのパスを DUMP_DEV に含めることをお勧めします。シャドウ・セットのメンバ変更が起きる頻度は、パスの変更が起きる頻度よりも少ないためです。

また、Alpha システムでは、DUMP_DEV に含むことができる以上のパスがある場合には、ダンプ・ディスクへのすべてのパスと、システム・ディスクへのできるだけ多くのパス (ただし少なくとも 1 つ) を定義することをお勧めします。システム・ディスクのパスは、リスト内の最後のエントリでなければならないことに注意してください。



注意:

システムのリブート時にエラー・ログ・バッファを復元できるように、エラー・ログ・ダンプ・ファイルは必ずシステム・ディスクに作成されます。このファイルは、DUMPSTYLE システム・パラメータや DUMP_DEV 環境変数の設定には影響されません。

システムは、リスト内で最初に見つかった有効なデバイスを、ダンプ・デバイスとして選択します。このため、ダンプ・ディスク・バス・エントリは、リスト内でシステム・ディスク・エントリよりも前に存在しなければなりません。

Alpha システムでは、DUMP_DEV に含めることができるデバイスの数に制限があります。この制限は、プラットフォームによって異なり、デバイスの構成によっても異なります。プラットフォームには、1 つのデバイスだけを使用できるものや、デバイスのリストを使用できるものがあります。デバイスのリストを使用できるシステムでは、リストの長さが、コンソール・サブシステムが使用する内部形式で 256 バイトに制限されています。この長さは、SCSI または CI ベース・ディスクを使用している場合はエントリ 4 個分の大きさ、ファイバ・チャネル・ディスクを使用している場合はエントリ 8 ~ 9 個分の大きさがあります。

I64 システムでは、プラットフォームやデバイスの構成に関わりなく、DUMP_DEV に最大 99 個のデバイスを含めることができます。

詳細は、後述の「I64 システムでのダンプ・デバイスの指定」を参照してください。

I64 システムで、次のいずれかの方法を使用してください。

- OpenVMS I64 Boot Manager (BOOT_OPTIONS.COM) ユーティリティを使用して、OpenVMS DCL プロンプトでダンプ・デバイス環境変数 DUMP_DEV を指定します。

```
$ @SYS$MANAGER:BOOT_OPTIONS
```

- または、EFI for OpenVMS (I64 のみ) の VMS_SET ユーティリティを使用して、EFI コンソール・プロンプトでダンプ・デバイス環境変数 DUMP_DEV を指定します。

```
Shell> FSN:\EFI\VMS\VMS_SET DUMP_DEV device-name[,...]
```

(FSn: VMS_SET ユーティリティが置かれている、ブート可能 FAT ファイル構造パーティション)

2.6.2 VAX システムでのシステム・ディスクへのシステム・ダンプ

システム・ディスクへのパスが2つ以上ある場合や、システム・ディスクのシャドウ・セットに複数のメンバがある場合に、確実にシステムがシステム・ディスクを検索し、そこにシステム・ダンプを書き込むことができるようにするには、プラットフォーム固有のブートに関する指示を守らなければなりません。正しいレジスタ値を設定しなければならない VAX システムがある一方で、特定の環境変数を設定しなければならない VAX システムもあります。詳細については、使用している VAX システムのアップグレードおよびインストールに関するマニュアルを参照してください。

システムに複数の CI スター・カブラがある場合には、シャドウ・セット・メンバはすべて同一のスター・カブラを経由して接続されていなければならないことに注意してください。

2.7 代替ディスクへのシステム・ダンプ・ファイルの書き込み

システム・ダンプ・ファイルは、OpenVMS システムのシステム・ディスク (DOSD) 以外のデバイスに書き込むことができます。大きいメモリを装備したシステムや、共通のシステム・ディスクを使用しているクラスタで、1つのディスクのディスク容量だけでは必要なダンプ・ファイルのサイズを必ずしもサポートできないクラスタでは、この機能は特に便利です。

DOSD を使用するための必要条件は、Alpha システムおよび I64 システムと、VAX システムとでは多少異なります。しかし、どのシステムでも、バグチェック・コードがシステム・ダンプ・ファイルを代替デバイスに書き込むことができるように、DUMPSTYLE システム・パラメータを正しく有効に設定しなければなりません。

以降の節では、Alpha システム、I64 システム、および VAX システムでの DOSD の必要条件について説明します。

2.7.1 Alpha システムおよび I64 システムでの DOSD の必要条件

Alpha システムおよび I64 システムでの DOSD の必要条件是次のとおりです。

- ダンプ・デバイスのディレクトリ構造は、現在のシステム・ディスク構造に類似していなければならない。[SYSn.SYSEXE]SYSDUMP.DMP ファイルはそこに存在し、同じブート時システム・ルートを持つ。

このファイルを作成するには、AUTOGEN を使用する。MODPARAMS.DAT ファイルで、次のシンボルは AUTOGEN に対してファイルを作成するように要求する。

```
DUMPFILDE_DEVICE = "$nnn$ddcuuuu"
```

デバイスのリストを入力できる。

- ダンプ・ディスクには ODS-2 または ODS-5 ファイル構造がなければならない。
- ダンプ・デバイスとしてボリューム・セットの一部を使用することはできない。
- 必須条件ではないが、システム・スタートアップ時にダンプ・デバイスをマウントするようにする。ダンプ・デバイスがマウントされると、CLUE と AUTOGEN はクラッシュ・ダンプ分析のためのアクセスができるようになる。最適な結果を得るために、MOUNT コマンドを SYS\$MANAGER:SYCONFIG.COM に含める。
- Crash Log Utility Extractor (CLUE) が DOSD をサポートできるためには、システム・クラッシュ後に分析されるファイルを論理名 CLUE\$DOSD_DEVICE として定義しておく必要がある。詳しくは 2.9 項「SDA CLUE コマンドによるクラッシュ・ダンプ・ファイルの分析 (Alpha および I64)」を参照。
- ダンプ・デバイスがシステム・ディスクであり、シャドウ・セットのマスタ・メンバである場合を除き、ダンプ・デバイスとしてシャドウ・セットの一部を使用することはできない。

- Alpha システムでは、コンソール・プロンプトに対してダンプ・デバイス環境変数 DUMP_DEV を指定するには、次の形式を使用する。
>>> SET DUMP_DEV device-name[,...]



注意:

DEC 3000 シリーズ・システムでは、DUMP_DEV 環境変数を使用するときに、次の制限事項が適用されます。

- DEC 3000 シリーズ・システムにはファイルの内容を保存するために十分な不揮発性 RAM がないため、システム電源障害が発生した後、この変数の内容は保存されません。したがって、電源障害が発生した後は、DUMP_DEV 変数を再設定しなければならない(しかし、他のタイプの再起動やブートストラップでは、DUMP_DEV は保存される)。
- DUMP_DEV をクリアすることはできない(ただし、システムの電源をオフにした後、再度電源を投入すれば、クリアすることができる)。
- 以前のバージョンでは DUMP_DEV がサポートされないため、コンソール・ファームウェア・バージョン 6.0 以上を使用しなければならない。

CPU タイプには、1つのデバイスだけを入力できるものや、デバイスのリストを入力できるものがある。リストにはシステム・ディスクとダンプ・ディスクに対するさまざまな代替パスを指定できる。

- I64 システムでは、OpenVMS I64 Boot Manager (BOOT_OPTIONS.COM) ユーティリティを使用して、OpenVMS DCL プロンプトでダンプ・デバイス環境変数 DUMP_DEV を指定する。

```
$ @SYS$MANAGER:BOOT_OPTIONS
```

または、EFI for OpenVMS (I64 のみ) の VMS_SET ユーティリティを使用して、EFI コンソール・プロンプトでダンプ・デバイス環境変数 DUMP_DEV を指定する。

```
Shell> FSn:\EFI\VMS\VMS_SET DUMP_DEV device-name[,...]
```

(FSn: VMS_SET ユーティリティが置かれている、ブート可能 FAT ファイル構造パーティション)

次の項を参照。

- DUMP_DEV を使用して代替パスを指定すると、システムの稼働時にディスクは代替パスにフェールオーバーできる。その後、システム・クラッシュが発生すると、バグチェック・コードは DUMP_DEV の内容を参照することにより、代替パスを使用できる。
- しかし、デバイスのリストを入力する場合には、システム・ディスクへのパスは最後に指定しなければならない。

Alpha システムでのダンプ・デバイスの指定

DUMP_DEV 環境変数を使用してダンプ・デバイスを指定するには、次の操作を実行します。

1. BOOTDEF_DEV の値を表示する。次の例を参照。

```
>>> SHOW BOOTDEF_DEV
```

```
BOOTDEF_DEV          dub204.7.0.4.3,dua204.4.0.2.3
```

2. 次に示すように、システムのデバイスを表示する。

```
>>> SHOW DEVICES
```

```
Resetting IO subsystem...
```

```

dua204.4.0.2.3      $4$DUA204 (RED70A)      RA72
dua206.4.0.2.3      $4$DUA206 (RED70A)      RA72
dua208.4.0.2.3      $4$DUA208 (RED70A)      RA72

```

polling for units on cixcd1, slot 4, xmi0...

```

dub204.7.0.4.3      $4$DUA204 (GRN70A)      RA72
dub206.7.0.4.3      $4$DUA206 (GRN70A)      RA72
dub208.7.0.4.3      $4$DUA208 (GRN70A)      RA72
>>>

```

この例で、次のことに注意する必要がある。

- DUA204 はシステム・ディスク・デバイスである。
- DUA208 は DOSD デバイスである。

3. システム・ディスクに対する 2 つのパスを指定し、ダンプ・ディスクを DUA208 として指定するには (やはり 2 つのパスを指定する)、次のように DUMP_DEV を設定する。

```
>>> SET DUMP_DEV dua208.4.0.2.3,dub208.7.0.4.3,dub204.7.0.4.3,dua204.4.0.2.3
```

この例で、dua208.4.0.2.3 と dub208.7.0.4.3 はダンプ・デバイスに対するパスである。dub204.7.0.4.3 と dua204.4.0.2.3 はブート・デバイスに対するパスである。

4. SHOW * コマンドを入力して、システムのすべての環境変数を表示する。次の例を参照。

```
>>> SHOW *
```

```

auto_action          HALT
baud                  9600
boot_dev              dua204.4.0.2.3
boot_file
boot_osflags          0,0
boot_reset            ON
bootdef_dev           dub204.7.0.4.3,dua204.4.0.2.3
booted_dev            dua204.4.0.2.3
booted_file
booted_osflags        0,0
cpu                   0
cpu_enabled            ff
cpu_primary            ff
d_harderr              halt
d_report               summary
d_softerr              continue
dump_dev              dua208.4.0.2.3,dub208.4.0.4.3,dub204.7.0.4.3,dua204.4.0.2.3
enable_audit           ON
interleave             default
language               36
pal                    V5.48-3/O1.35-2
prompt                 >>>
stored_argc            2
stored_argv0           B
stored_argv1           dua204.4.0.2.3
system_variant         0
version                T4.3-4740 Jun 14 2003 15:16:38
>>>

```

この項の手順が完了したら、次の項の後にある 2.7.1 項「Alpha システムと I64 システムでの DOSD の有効化」を参照してください。

I64 システムでのダンプ・デバイスの指定

I64 システムのダンプ・デバイスを指定するには、次のいずれかを使用します。

- Boot Manager ユーティリティ、BOOT_OPTIONS.COM。
- EFI ユーティリティ・コマンド VMS_SET。



注記: OpenVMS I64 Boot Manager Utility を使用することをお勧めします。(このユーティリティの使用は、ファイバ・チャネル以外のデバイスではオプションですが、ファイバ・チャネル・デバイスでは必須です。) このユーティリティについての詳細は、『OpenVMS システム管理者マニュアル(上巻)』を参照してください。

Boot Manager ユーティリティ BOOT_OPTIONS.COM の使用

OpenVMS DCL プロンプトから DUMP_DEV 環境変数でダンプ・デバイスを指定するには、次の手順に従います。

1. DCL プロンプトで次のコマンドを入力して、OpenVMS I64 Boot Manager ユーティリティを起動する。

```
$ @SYS$MANAGER:BOOT_OPTIONS.COM
```

2. ユーティリティが起動され、メイン・メニューが表示される。プロンプトで D を入力して、ダンプ・デバイスオプション・リストに対して操作を行うように設定する。

```
Enter your choice: D
```

ダンプ・オプション・リストへエントリを追加したときの出力例を、次に示す。

```
OpenVMS I64 Boot Manager Dump Options List Management Utilities
```

- (1) ADD an entry to the Dump Options list
- (2) DISPLAY the Dump Options list
- (3) REMOVE an entry from the Dump Options list
- (4) MOVE the position of an entry in the Dump Options list

- (B) Set to operate on the Boot Device Options list
- (D) Set to operate on the Dump Device Options list
- (G) Set to operate on the Debug Device Options list

(E) EXIT from the Boot Manager utility

You can also enter Ctrl-Y at any time to abort this utility.



注記: このユーティリティを使用している間は、必要なだけ上矢印 (^) を入力することで、応答内容を変更することができる。このプログラムを終了して DCL プロンプトに戻るには、Ctrl/Y を入力する。

3. DUMP_DEV 環境変数にダンプ・デバイスを設定するオプションを指定するには、次のプロンプトで 1 を入力する。

```
Enter your choice: 1
```

4. ユーティリティは、デバイス名の入力を求めるプロンプトを表示する。次の例のように、DOSD デバイスのデバイス名を入力する。この例のダンプ・デバイスは、マルチパス・ファイバ・デバイス DGA1 である。

```
Enter the device name (Enter "?" for a list of devices): $1$DGA1
```

```
efi$bcfg: $1$DGA1 (VMS_DUMP_DEV_01) Option successfully added
efi$bcfg: $1$DGA1 (VMS_DUMP_DEV_02) Option successfully added
efi$bcfg: $1$DGA1 (VMS_DUMP_DEV_03) Option successfully added
```

5. 他の DUMP_DEV デバイスも追加する場合は、手順 3 と 4 を繰り返す。
6. すべての DUMP_DEV オプションが正常に追加されたら、次のプロンプトで E を入力して、ユーティリティから抜ける。

Enter your choice: **E**

7. システムをリブートし、「Alpha システムと I64 システムでの DOSD の有効化」の項に進む。

OpenVMS 用の EFI ユーティリティ VMS_SET コマンドの使用

OpenVMS 用の EFI ユーティリティ (I64 のみ) を使用して、EFI コンソール・プロンプトで、ダンプ・デバイス環境変数 DUMP_DEV を指定します。

1. システム上のデバイスを、次のように表示する。

```
Shell> FS0:\EFI\VMS\VMS_SHOW DEVICE

VMS: EIA0 0-30-6E-39-F7-A5
EFI: Acpi(000222F0,0)/Pci(3|0)/Mac(00306E39F7A5
VMS: DKA0 HP 18.2GATLAS10K3_18_SCAHP05
EFI: Acpi(000222F0,100)/Pci(1|0)/Scsi(Pun0,Lun0)
VMS: DKB400 HP 18.2GST318406LC HP05
EFI: fs2: Acpi(000222F0,100)/Pci(1|1)/Scsi(Pun4,Lun0)
VMS: DKB200 HP 18.2GST318406LC HP05
EFI: fs1: Acpi(000222F0,100)/Pci(1|1)/Scsi(Pun2,Lun0)
VMS: DKB0 HP 18.2GATLAS10K3_18_SCAHP05
EFI: fs0: Acpi(000222F0,100)/Pci(1|1)/Scsi(Pun0,Lun0)
VMS: EWA0 0-30-6E-39-77-3
EFI: Acpi(000222F0,100)/Pci(2|0)/Mac(00306E39773D
```

この例では、次のことが分かる。

- DKB0 と DKB200 は、システム・ディスクのシャドウ・セットのメンバである。
- DKA0 は、DOSD デバイスである。

2. DUMP_DEV を、次のように設定する。

```
Shell> FS0:\EFI\VMS\VMS_SET DKA0, DKB0, DKB200

VMS: DKA0 HP 18.2GATLAS10K3_18_SCAHP05
EFI: Acpi(000222F0,100)/Pci(1|0)/Scsi(Pun0,Lun0)
VMS: DKB0 HP 18.2GATLAS10K3_18_SCAHP05
EFI: fs0: Acpi(000222F0,100)/Pci(1|1)/Scsi(Pun0,Lun0)
VMS: DKB200 HP 18.2GST318406LC HP05
EFI: fs1: Acpi(000222F0,100)/Pci(1|1)/Scsi(Pun2,Lun0)
```

この項の作業を完了したら、次の項を参照してください。

Alpha システムと I64 システムでの DOSD の有効化

最後に、DUMPSTYLE システム・パラメータのビット 2 をセットすることにより、DOSD ビットを有効にします。たとえば、SYSBOOT> プロンプトに対して 4 という値を入力すると、圧縮されていない物理ダンプを代替ディスクに書き込み、コンソール出力をできるだけ少なくすることを指定できます。

```
SYSBOOT> SET DUMPSTYLE 4
```

DUMPSTYLE システム・パラメータについての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』、およびオンライン・ヘルプを参照してください。



注意:

システムのリブート時にエラー・ログ・バッファを復元できるように、エラー・ログ・ダンプ・ファイルはシステム・ディスクに必ず作成されます。このファイルは、DUMPSTYLE システム・パラメータの設定や、DUMP_DEV 環境変数の設定の影響を受けません。

システムは、リスト内で最初に見つかった有効なデバイスを、ダンプ・デバイスとして選択します。このため、ダンプ・ディスク・パス・エントリは、リスト内でシステム・ディスク・エントリよりも前に存在しなければなりません。

Alpha システムでは、DUMP_DEV に含めることができるデバイスの数に制限があります。この制限は、プラットフォームによって異なり、デバイスの構成によっても異なります。プラットフォームには、1 つのデバイスだけを使用できるものや、デバイスのリストを使用できるものがあります。デバイスのリストを使用できるシステムでは、リストの長さが、コンソール・サブシステムが使用する内部形式で 256 バイトに制限されています。この長さは、SCSI または CI ベース・ディスクを使用している場合はエントリ 4 個分の大きさ、ファイバ・チャネル・ディスクを使用している場合はエントリ 8 ~ 9 個分の大きさがあります。

164 システムでは、プラットフォームやデバイスの構成に関わりなく、DUMP_DEV に最大 99 個のデバイスを含めることができます。

2.7.2 VAX システムでの DOSD の必要条件

VAX システムでの DOSD の必要条件は次のとおりです。

- システムは CI コントローラに直接接続されていなければならない。CI コントローラからブートしなければならない。
- ダンプ・デバイスはブート・デバイスと同じ 2 台の HS_x CI コントローラに物理的に接続しなければならない。これら 2 台のコントローラは、すべて同一のスター・カプラを経由して接続されていなければならない。
- ダンプ・デバイスのディレクトリ構造は、現在のシステム・ディスクの構造と類似していなければならない。[SYSn.SYSEX]SYSDUMP.DMP ファイルはそこに存在し、同じブート時システム・ルートを使用する。

このファイルを作成するには、AUTOGEN を使用する。MODPARAMS.DAT ファイルで次のシンボルは、AUTOGEN に対してファイルを作成するように要求する。

```
DUMPFILE_DEVICE = "$nnn$ddcuuuu"
```

1 つのデバイスだけを指定できる。

- ボリューム・ラベルは 12 文字以内の長さである。このボリューム・ラベルの一部として、ASCII 文字列 DOSD_DUMP を指定しなければならない。たとえば、DOSD_DUMP、DOSD_DUMP_12、12_DOSD_DUMP はすべて正しいボリューム・ラベルである。ラベルは読み込まれ、メモリ・ブート・データ構造体に格納される。
- ダンプ・デバイスをボリューム・セットの一部として使用することはできない。また、ダンプ・デバイスをシャドウ・セットの一部として使用することも望ましくない。
- ダンプ・デバイスとして MSCP ユニット・ゼロ (0) を使用することはできない。ユニット 1 ~ 4095 (1 ~ FFF) だけがサポートされる。

ダンプ・デバイスは次のように指定できる。

- VAX 7000 構成では、DUMPSTYLE システム・パラメータのビット 16 ~ 27 を使用する。VAX 7000 構成で提供される DUMP_DEV 環境変数は、OpenVMS VAX では使用されないことに注意しなければならない。
- VAX 7000 以外の構成では、レジスタ 3 (R3) のビット 16 ~ 27 を使用する。レジスタのこの部分を使用して、ダンプ・デバイスを指定できる。

DUMPSTYLE システム・パラメータについての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』、およびオンライン・ヘルプを参照。



注意:

システム・クラッシュが発生した後でシステムをリブートするときにエラー・ログ・バッファを復元するには、エラー・ログをシステム・ディスクに保存しておかなければなりません。AUTOGEN は、そのためにシステム・ディスクに SYSDUMP.DMP ファイルを作成します。このファイルは、エラー・ログ・バッファの最大サイズを格納できるだけの十分な大きさです。

2.8 SDA によるクラッシュ・ダンプの内容の分析

システム・ダンプ・アナライザ・ユーティリティ (SDA) を使用してシステム・ダンプ・ファイルの内容を翻訳し、クラッシュの予想される原因を調べることができます。クラッシュ・ダンプの分析については、『OpenVMS VAX System Dump Analyzer Utility Manual』または『OpenVMS Alpha System Analysis Tools Manual』を参照してください。

システムに障害が発生した場合は、SDA を使用して障害発生時のシステム・ダンプ・ファイルのコピーを作成し、弊社のサポート担当者に連絡してください。システム・ダンプ・ファイルのコピーの作成については、2.12 項「システム・ダンプ・ファイルをテープまたはディスクへコピーする」を参照してください。

2.9 SDA CLUE コマンドによるクラッシュ・ダンプ・ファイルの分析 (Alpha および I64)

SDA CLUE (Crash Log Utility Extractor) コマンドは、クラッシュ・ダンプの分析と、スタンドアロン・システムやクラスタで発生した重大なバグのチェックの履歴の管理を自動的に行います。SDA CLUE コマンドは、SDA とともに使用し、標準の SDA からアクセス困難なダンプ・ファイル補足情報を収集およびデコードすることができます。また、SDA CLUE コマンドを、Dump Off System Disk (DOSD) とともに使用し、システム・ディスク以外のディスクにあるシステム・ダンプ・ファイルを解析することができます。

2.9.1 CLUE について (Alpha および I64)

Alpha システムおよび I64 システムでは、システム障害後にシステムをリブートするとき、SDA は自動的に呼び出されます (省略時の設定)。クラッシュ・ダンプの分析をより容易にするために、SDA CLUE コマンドは自動的に CLUE リスト・ファイルのダンプ・ファイル要約情報を取得および保管します。

スタートアップ・コマンド・プロシージャは、次の動作を行うコマンドを起動します。

- SDA の呼び出し
- SDA CLUE HISTORY コマンドの発行
- CLUE\$nodename_ddmmyy_hhmm.LIS リスト・ファイルの作成

CLUE HISTORY コマンドは、履歴ファイルに要約エントリを 1 行だけ追加し、SDA CLUE コマンドの次の出力をリスト・ファイルに保存します。

- クラッシュ・ダンプ要約情報
- システム構成
- スタック・デコーダ
- ページ・ファイルとスワップ・ファイル
- メモリ管理統計情報
- プロセス DCL 再呼び出しバッファ
- アクティブ XQP プロセス
- XQP キャッシュ・ヘッダ

この CLUE リスト・ファイルの内容は、システム障害を分析するときに便利です。

このようなファイルがしきい値 (省略時 5000 ブロック) を超えるまで蓄積される場合、(しきい値の限界内になるまで) 古いファイルから削除されます。CLUE\$MAX_BLOCK 論理名を使って、これをカスタマイズすることも可能です。

システムのスタートアップ時に CLUE を実行しないようにするには、SYLOGICALS.COM ファイル中の論理名 CLUE\$INHIBIT を /SYS TRUE と定義します。

CLUE\$nodename_ddmmyy_hhmm.lis にはクラッシュ・ダンプの概要しか入っておらず、常にクラッシュの原因を決定するのに十分な情報が入っているとは限りませんので注意してください。システム・クラッシュの分析を詳細に行わなければならない場合は、常に SDA COPY コマンドを使用して、ダンプ・ファイルを保存しておくことをお勧めします。

2.9.2 SDA CLUE コマンドによるデータの表示 (Alpha および I64)

次のように、SDA プロンプトから CLUE コマンドを呼び出します。

```
SDA> CLUE CONFIG
```

CLUE コマンドは、ダンプ・ファイルから取得したクラッシュ・ダンプの要約情報を提供します。クラッシュ・ダンプを会話形式でデバッグするとき、SDA CLUE コマンドを使って、ダンプ・ファイルから取得情報を収集およびデコードすることができます。このファイルは標準の SDA からでは簡単にアクセスすることができません。一方、CLUE は詳細な XQP 要約を即座に提供します。

実行中のシステム上で CLUE コマンドを会話形式で使用して、性能問題を識別することも可能です。

クラッシュ・ダンプを分析するときは、すべての CLUE コマンドが使用できます。しかし、実行中のシステムを分析するときには、CLUE コマンドの CLUE CRASH、CLUE ERRLOG、CLUE HISTORY、および CLUE STACK は使用できません。

SDA CLUE コマンドの使用について詳しくは、『OpenVMS Alpha System Analysis Tools Manual』を参照してください。

2.9.3 ダンプ・オフ・システム・ディスクと SDA CLUE の使用 (Alpha および I64)

ダンプ・オフ・システム・ディスク (DOSD) によって、システム・ダンプ・ファイルをシステム・ディスク以外のデバイスに書き出すことができます。システム・クラッシュが発生した後、SDA CLUE が解析するダンプ・ファイルを正しくみつけるために、次の操作を実行します。

1. コマンド・プロシージャ SYS\$MANAGER:SYCONFIG.COM にダンプ・ファイルを指すシステム論理名 CLUE\$DOSD_DEVICE を加える。ファイル指定なしで物理デバイス名または論理デバイス名だけを指定する。
2. ダンプ・ファイルのあるデバイスをシステム単位のデバイスとしてマウントするように SYS\$MANAGER:SYCONFIG.COM へコマンドを追加する。これを行わないと、SDA CLUE はダンプ・ファイルにアクセスして解析をおこなうことができない。

次の例では、ダンプ・ファイルは デバイス \$3\$DUA25 にあり、ラベルは DMP\$DEV です。次のコマンドを SYS\$MANAGER:SYCONFIG.COM に追加しておく必要があります。

```
$ mount/system/noassist $3$dua25: dmp$dev dmp$dev  
$ define/system clue$dosd_device dmp$dev
```

2.10 CLUE を使用して、クラッシュ・ダンプに関する履歴情報を得る方法 (VAX のみ)

VAX システムにおいて、クラッシュ・ログ・ユーティリティ・エキストラクタ (CLUE) は、**クラッシュ履歴ファイル**の内容を表示するために使用できるツールです。クラッシュ履歴ファイルの内容を調べることにより、障害(クラッシュ)の原因となった問題を理解して解決することができる場合があります。また、場合によってはその他の有用なデータを得ることもできます。

2.10.1 CLUE について (VAX のみ)

クラッシュ履歴ファイルは、CLUE により作成、更新されるもので、主なパラメータはクラッシュ・ダンプ・ファイルからとります。システムの障害が発生するたびに書き換えられ、そのため、通常は最も新しい障害に対してしか使用できないクラッシュ・ダンプとは異なり、クラッシュ履歴ファイルはシステム障害の永久的な記録です。

システムに障害が発生し、物理メモリがクラッシュ・ダンプ・ファイルにコピーされると、CLUE はシステムのリスタート時、関連するパラメータを自動的にファイル CLUE\$OUTPUT:CLUE\$HISTORY.DATA に付加します。ここでは、CLUE を使用して収集したデータを CLUE を使って表示する方法について説明します。CLUE については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。



注意:

履歴ファイルは各エントリごとに通常 10 から 15 ブロックほどの割合で大きくなっていきます。バイナリ・ファイル内のエントリ数は、論理名 CLUE\$MAX_ENTRIES を必要な最大数になるように定義することにより、制限することができます。この最大数に達すると、最も古いエントリが履歴ファイルから削除されます。

省略時の設定では、オペレータによるシャットダウンは履歴ファイルに記録されます。論理名 CLUE\$EXCLUDE_OPERS を TRUE として定義することにより、オペレータによるシャットダウンに関する情報を履歴ファイルから削除することができます。たとえば、SYS\$MANAGER:SYSTARTUP_VMS.COM に次の行を含めます。

```
$ DEFINE /SYSTEM CLUE$EXCLUDE_OPERS TRUE
```

2.10.2 CLUE を使用したデータの表示 (VAX のみ)

CLUE を使用してデータを表示するには、まず、次のシンボルを定義する必要があります。

```
$ CLUE := $CLUE
```

シンボルを定義したら、次のコマンドを入力することにより、CLUE を使って情報を表示することができます。

```
$ CLUE/DISPLAY
```

```
CLUE_code_example>
```

プロンプト CLUE_code_example> に対してコマンドを発行し、次の作業を行うことができます。

- DIRECTORY コマンドを使用して、指定日付以降発生した障害、特殊なタイプの障害、指定モジュールを含む障害、指定オフセットを持つ障害をリストする。
たとえば、次のように、DIRECTORY コマンドを使用して全障害を履歴ファイルにリストすることができる。

```
CLUE_code_example> DIRECTORY
```

- SHOW コマンドを使用して、システム・ダンプ・アナライザ (SDA) の特定のコマンドから得たものと同様の情報を生成する。

たとえば、クラッシュ番号 7 でリストされたクラッシュに関する全情報が必要な場合は、次の SHOW コマンドを使用する。

```
CLUE_code_example> SHOW ALL 7
```

- EXTRACT コマンドを使用して、エントリから得たデータをファイルに書き込む。
たとえば、次のコマンドにより、クラッシュ履歴ファイルのエントリ番号 7 から得たデータを 15MAYCRASH.TXT という名前のファイルに書き込む。

```
CLUE_code_example> EXTRACT 7/OUTPUT=15MAYCRASH.TXT
```

CLUE コマンドについては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』を参照してください。

2.11 システム障害後のシステム・ダンプ・ファイルの内容の保存

システム障害が発生すると、クラッシュ・ダンプ・ファイルの内容は上書きされ、以前の内容が失われます。したがって、システムをリブートするたびに、システム・ダンプ・ファイルを自動的に分析し、コピーするように設定しておく必要があります。

Alpha システムおよび I64 システムでは、スタートアップ時に SDA が呼び出され (省略時の設定)、CLUE リスト・ファイルが作成されます。CLUE リスト・ファイルは一連のコマンドで作成され、クラッシュの概要だけが含まれます。クラッシュの原因を決定するのに十分な情報を持っていません。したがって、常にダンプ・ファイルをコピーしておくことをお勧めします。

ユーザのサイト別コマンド・プロシージャに、システム障害後のスタートアップ時に実行させるコマンド (SDA COPY など) を追加する方法については、『OpenVMS Alpha System Analysis Tools Manual』を参照してください。

VAX システムでは、システムのブート時にシステム・ダンプ分析ユーティリティ (SDA) を呼び出すように、サイト別スタートアップ・コマンド・プロシージャ SYSTARTUP_VMS.COM を変更します。

次の点に注意してください。

- STARTUP プロセスでサイト別スタートアップ・プロシージャから実行されている場合、システム障害の直後にシステムがブートしているときのみ、SDA は指定されたコマンドを実行する。SHUTDOWN.COM または OPCCRASH.EXE でシャットダウンされた後のリブートの場合、SDA は、コマンドを実行しないで終了する。
- ダンプ・ファイルをコピーするときは、DCL の COPY コマンドを使用できるが、できるだけ SDA の COPY コマンドを使用する。SDA の COPY コマンドはダンプによって占有されたブロックだけをコピーし、コピーしたダンプ・ファイルをマークするためである。SDA の COPY コマンドは、ダンプがページング・ファイル SYS\$SYSTEM:PAGEFILE.SYS に書き込まれたときにも使用したほうがよい。SDA の COPY コマンドが、ダンプによって占有されているページをページャに解放するためである。詳細は2.13 項「ページ・ファイルからのダンプ情報の解放」を参照。
- システム・ダンプ・ファイルには、特権情報が含まれている場合があるので、ダンプ・ファイルは一般ユーザにアクセスされないよう保護しておく。ファイル保護についての詳細は、『OpenVMS システム・セキュリティ・ガイド』を参照。
- システム・ダンプ・ファイルには NOBACKUP 属性があるため、バックアップ・ユーティリティ (BACKUP) は、BACKUP の起動時に、修飾子 /IGNORE=NOBACKUP を使用しない限り、それらをコピーしない。SDA COPY コマンドを使用して、システム・ダンプ・ファイルを別のファイルにコピーする場合、オペレーティング・システムにより新しいファイルが自動的に NOBACKUP に設定されることはない。コピーに NOBACKUP 属性を設定したい場合は、『OpenVMS DCL ディクショナリ』で説明しているように、SET FILE コマンドに /NOBACKUP 修飾子を指定する。

例

この例では、SDA の COPY コマンドを使って SYS\$SYSTEM:PAGEFILE.SYS ファイルの内容を保存し、そのファイルを分析しています。ダンプで使用されるページ・ファイルは COPY コマンドが完了するとすぐに解放され、別の SDA コマンドが実行される前にページングに使用されることがあるため、この COPY コマンドは最終コマンドになることに注意してください。

```
$ !
$ !      Print dump listing if system just failed
$ !
$ ANALYZE/CRASH_DUMP SYS$SYSTEM:PAGEFILE.SYS
```

```

SET OUTPUT DISK1:SYSDUMP.LIS          ! Create listing file
READ/EXECUTIVE                        ! Read in symbols for kernel
SHOW CRASH                            ! Display crash information
SHOW STACK                            ! Show current stack
SHOW SUMMARY                          ! List all active processes
SHOW PROCESS/PCB/PHD/REG              ! Display current process
COPY SYS$SYSTEM:SAVEDUMP.DMP         ! Save system dump file
EXIT

```

```

$ SET FILE/NOBACKUP SYS$SYSTEM:SAVEDUMP.DMP

```

2.12 システム・ダンプ・ファイルをテープまたはディスクへコピーする

システム障害が発生した場合は、システム・ダンプ・ファイルの内容のコピーを作成して、弊社のサポート担当者にお知らせください。バックアップ・ユーティリティ (BACKUP) を使用すると、システム・ダンプ・ファイルを含むセーブ・セットを、磁気テープまたはディスクに作成できます。ただし、BACKUP を使用してシステム・ダンプ・ファイルをコピーする場合は、次の理由から、/IGNORE=(NOBACKUP,INTERLOCK) 修飾子を指定する必要があります。

- 省略時の設定では、システム・ダンプ・ファイルには NOBACKUP 属性がある。したがって、/IGNORE=NOBACKUP を指定しない限りコピーされない。
- システムには、ダンプ・ファイルへのオープン・チャンネルが保持されている。したがって、/IGNORE=INTERLOCK を指定しない限り、ファイルはコピーされない。

BACKUP の使用方法については、『OpenVMS システム管理者マニュアル (上巻)』を参照してください。また、BACKUP コマンドについては、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (上巻)』の BACKUP の節を参照してください。

システム・ダンプ・ファイルをコピーする場合は、できるだけ次の手順に従ってください。

1. SDA COPY コマンドを使用して、システム・ダンプ・ファイルのコピーを作成する。
2. BACKUP ユーティリティを使用して、作成したコピーをテープまたはディスクに保存する。

この方法で行えば、BACKUP の修飾子による問題を回避できます。また SDA COPY コマンドは、実際に使用されているブロックだけをシステム・ダンプ・ファイルにコピーするので、テープに書き込まれるデータ量を減らすことができます。

2.13 ページ・ファイルからのダンプ情報の解放

システム・クラッシュ・ダンプを SYS\$SYSTEM:PAGEFILE.SYS に保存した場合、ページャのため、ダンプに占有された空間を後で解放する必要があります。そうしないと、ページング空間が不足してシステムがハングすることがあります。

どんな場合に、システム・クラッシュ・ダンプをページ・ファイルに保存するのかについては 2.1.1 項「ページ・ファイルを使ってシステム・クラッシュ・ダンプを格納する方法」を参照してください。

2.13.1 VAX システム, Alpha システム, および I64 システムでのダンプ情報の解放

ここでは、VAX システム, Alpha システム, および I64 システムでページ・ファイルからダンプ情報を解放する方法について説明します。

作業方法

VAX システムの場合は、次の手順に従ってください。

1. PAGEFILE.SYS をターゲットにして、システム・ダンプ・アナライザ・ユーティリティ (SDA) を起動する。

```

$ ANALYZE/CRASH_DUMP SYS$SYSTEM:PAGEFILE.SYS

```

2. SDA の COPY コマンドを次の形式で入力し、ダンプを SYS\$SYSTEM:PAGEFILE.SYS から別のファイルにコピーする。
COPY ダンプ・ファイル指定
たとえば、ダンプ・ファイルをシステム・ディスクから DISK\$USER5 上の SAVEDUMP.DMP というファイルにコピーするには、次のコマンドを入力する。

```
SDA> COPY DISK$USER5:[DUMPS] SAVEDUMP.DMP
```

3. EXIT コマンドで SDA を終了する。
4. ステップ 1 および ステップ 2 で入力した SDA コマンドを、サイト別スタートアップ・コマンド・プロシージャ SYSTARTUP_VMS.COM に追加する。これにより、システムがリブートするたびにページ空間が解放されるようになる。

また、ページング・ファイルのダンプが格納されているページを他にコピーしないで解放するには、ANALYZE/CRASH_DUMP/RELEASE コマンドを使用します。このコマンドは、ダンプを効果的に削除し、システム・ページングのために使用されるページをただちに解放します。ただし、このコマンドでは、削除を行う前にダンプを分析することは**できません**。

例

次のコマンドは、SYSTARTUP_VMS.COM コマンド・プロシージャに追加され、システムのリブート時に、ページ・ファイルの内容を SAVEDUMP.DMP というファイルにコピーします。

```
$ ANALYZE/CRASH_DUMP SYS$SYSTEM:PAGEFILE.SYS
COPY DISK$USER5:[DUMPS] SAVEDUMP.DMP
EXIT
$ SET FILE/NOBACKUP SYS$SYSTEM:SAVEDUMP.DMP
```

Alpha システムおよび I64 システムでの作業方法

Alpha システムおよび I64 システムでは、『OpenVMS Alpha System Analysis Tools Manual』で説明しているように、システム障害が発生した後でシステムをリブートするときに、省略時の設定により SDA は自動的に起動されます。

システム・ダンプ・ファイルを自動的に保存するには、次の操作を実行します。

1. SYS\$MANAGER:SAVEDUMP.COM ファイルを作成する。次の例を参照。

```
!
! SDA command file, to be executed as part of the system
! bootstrap from within CLUE.  Commands in this file can
! be used to save the dump file after a system bugcheck, and
! to execute any additional SDA command.
!
READ/EXEC           ! Read in the executive images' symbol tables
SHOW STACK         ! Display the stack
COPY SAVEDUMP.DMP  ! Copy and save system dump file
```

2. 使用しているシステム固有のファイルを指すには、次のような行をファイル SYS\$MANAGER:SYLOGICALS.COM に追加する。

```
$ DEFINE/SYSTEM CLUE$SITE_PROC SYS$MANAGER:SAVEDUMP.COM
```

この例では、使用しているシステム固有のファイルは SAVEDUMP.COM という名前である。

論理名 CLUE\$INHIBIT が定義されており、システム・スタートアップ時に SDA が自動的に起動されなかった場合には、ダンプによって使用されたページ・ファイル内のページは、ANALYZE/CRASH_DUMP/RELEASE コマンドを使用して解放できます。このコマンドは、ダン

ブを効果的に削除し、システム・ページングのために使用されるページをただちに解放します。このコマンドでは、ダンプを削除する前に分析することは**できません**。

CLUE が使用する論理名については『OpenVMS Alpha System Analysis Tools Manual』を参照してください。

2.13.2 VAX システム、Alpha システム、および I64 システムでダンプ情報を解放するための使用上の注意

システム・ダンプ・ファイルには特権情報が格納されている可能性があるため、ダンプ・ファイルのコピーがワールドから読み込みされないように保護してください。

システムがファイルの内容全体をバックアップしないようにするには、DCL の SET FILE/NOBACKUP コマンドを使用して、ファイルに NOBACKUP 属性を割り当てます。

また、DCL の COPY コマンドを使用してダンプ・ファイルをコピーすることもできますが、なるべく SDA の COPY コマンドを使用してください。これは、SDA COPY コマンドが次の操作を実行するからです。

- ダンプで実際に占有しているブロックだけをコピーする。
- システムのページ・ファイルでダンプが占有しているページを、ページングするために解放する。

2.14 ページ・ファイルとスワップ・ファイルのインストール

SYS\$SYSTEM にある 1 次ページ・ファイルと 1 次スワップ・ファイルは、システムによって自動的にインストールされます。ただし、その他のファイルが自動的にインストールされることはありません。このため、2 次ページ・ファイルまたは 2 次スワップ・ファイルを作成した場合には、システム生成ユーティリティ (SYSGEN) を使って、それらをインストールする必要があります。SYSGEN の INSTALL コマンドは、INSTALL ユーティリティ・コマンドとは異なる働きをする点に注意してください。

2.14.1 会話型のインストール

1. 次のコマンドを入力して、SYSGEN を起動する。

```
§ RUN SYS$SYSTEM:SYSGEN
```

2. 次のように、SYSGEN の INSTALL コマンドを入力する。

ページ・ファイルの場合は、次の形式で入力する。

```
INSTALL ファイル指定/PAGEFILE
```

次の例を参照。

```
SYSGEN> INSTALL DUA2:[PAGE_SWAP]PAGEFILE_1.SYS/PAGEFILE
```

スワップ・ファイルの場合は、次の形式で入力する。

```
INSTALL ファイル指定/SWAPFILE
```

次の例を参照。

```
SYSGEN> INSTALL DUA2:[PAGE_SWAP]SWAPFILE_1.SYS/SWAPFILE
```

3. システムがブートするたびにファイルがインストールされるように、SYS\$MANAGER:SYPAGSWPFILES.COM にステップ 2 で入力したコマンドを追加する。詳細は 2.14.2 項「SYPAGSWPFILES.COM でのインストール」を参照。

例

ページ・ファイルおよびスワップ・ファイルを会話形式でインストールします。

```
§ RUN SYS$SYSTEM:SYSGEN
```

```
SYSGEN> INSTALL DUA2:[PAGE_SWAP]PAGEFILE_1.SYS/PAGEFILE
SYSGEN> INSTALL DUA2:[PAGE_SWAP]SWAPFILE_1.SYS/SWAPFILE
```

2.14.2 SYPAGSWPFILES.COM でのインストール

SYS\$SYSTEM:PAGEFILE.SYS, SYS\$SYSTEM:SWAPFILE.SYS 以外のページ・ファイルおよびスワップ・ファイルは、システムがブートするたびにインストールしなおす必要があります。この作業は、スタートアップ・コマンド・プロシージャ SYS\$MANAGER:SYPAGSWPFILES.COM にファイルをインストールするコマンドを追加することによって自動化できます。テンプレート・ファイル SYS\$MANAGER:SYPAGSWPFILES.TEMPLATE には、このファイルがどのように使用されているかを説明するコメントが含まれます。

この作業を行う前には、2.16 項「ページ・ファイル、スワップ・ファイル、ダンプ・ファイルの作成と変更」で説明したように、2 次ファイルを作成しておく必要があります。

SYPAGSWPFILES.COM についての詳細は、『OpenVMS システム管理者マニュアル (上巻)』を参照してください。

また、SATELLITE_PAGE.COM を使って、OpenVMS Cluster サテライト・ノードのローカル・ディスク上に、ページ・ファイルおよびスワップ・ファイルをインストールすることもできます。SATELLITE_PAGE.COM は、CLUSTER_CONFIG.COM を実行すると作成されます。サテライト・ノードのローカル・ディスク上にページ・ファイルおよびスワップ・ファイルをインストールする方法については、『OpenVMS Cluster システム』を参照してください。

作業方法

1. SYS\$MANAGER:SYPAGSWPFILES.COM を編集するために、任意のエディタを起動する。
2. 必要に応じて、ページ・ファイルおよびスワップ・ファイルを保持するディスクをマウントするための MOUNT コマンドを追加する。この作業が必要なのは、SYPAGSWPFILES.COM の起動時には、システム・ディスクしかインストールされていないためである。

例：

```
$ MOUNT/SYSTEM/NOASSIST DUA2: DISK_SYS2
```

MOUNT コマンドについては、『OpenVMS DCL デクシヨナリ』を参照。

次のコマンドは、MOUNT コマンドの前に挿入する。このコマンドも、マウント前にディスクが使用できるかどうかを判断するのに有用である。ただし、ディスクが破損していてマウントできない場合は、これらのコマンドにより無限ループが発生する。

```
$ LOOP1:
$ ON WARNING THEN GOTO LOOP1
$ WAIT 0000 00:00:00.50
$ READY = F$GETDVI("device:", "AVL")
$ IF READY .EQS. "FALSE" THEN GOTO LOOP1
```

ここで、*device:* には、デバイス名を指定する。

3. SYSGEN を起動する次のコマンドを追加する。

```
$ RUN SYS$SYSTEM:SYSGEN
```

4. 次の形式のコマンドを SYPAGSWPFILES.COM に追加する。これらのコマンドは、システムがブートするたびに該当するファイルをインストールする。

ページ・ファイルの場合は、次の形式を使用する。

```
INSTALL ファイル指定/PAGEFILE
```

例:

```
INSTALL DUA2:[SYSTEM]PAGEFILE_1.SYS/PAGEFILE
```

スワップ・ファイルの場合は、次の形式を使用する。

INSTALL ファイル指定/SWAPFILE

例:

```
INSTALL DUA2:[SYSTEM] SWAPFILE_1.SYS/SWAPFILE
```

5. EXIT コマンドを追加して SYSGEN を終了する。

```
EXIT
```

例

デバイス DUA2: 上に存在する、PAGEFILE_1.SYS および SWAPFILE_1.SYS という名前のファイルをインストールする場合には、SYPAGSWPFILES.COM に追加するコマンドを示します。

```
$ EDIT SYS$MANAGER:SYPAGSWPFILES.COM  
[add the following commands to SYPAGSWPFILES.COM:]
```

```
$ MOUNT/SYSTEM/NOASSIST DUA2: DISK_SYS2  
$ RUN SYS$SYSTEM:SYSGEN  
INSTALL DUA2:[SYSTEM] PAGEFILE_1.SYS /PAGEFILE  
INSTALL DUA2:[SYSTEM] SWAPFILE_1.SYS /SWAPFILE  
EXIT
```

2.15 ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの削除



重要:

ページ・ファイル, スワップ・ファイル, ダンプ・ファイルを削除する場合には、単にファイルだけを削除しないでください。ファイルだけを削除した後、システム・ディスクを継続して使用していると、システム・ディスクが破損することがあります。

作業方法

1. RENAME コマンドを使って、削除するファイルの名前を変更する。
2. システムをシャットダウンし、リブートする。
3. ファイルを削除する。
4. ファイルを削除する場合は、SYPAGSWPFILES.COM および MODPARAMS.DAT から、そのファイルに関連するすべてのコマンド行を確実に削除するようにする。

例

```
$ RENAME DUA2:[SYSTEM] PAGEFILE_1.SYS; DUA2:[SYSTEM] JUNK.SYS;  
$ @SYS$SYSTEM:SHUTDOWN.COM
```

```
[SHUTDOWN.COM shuts down and reboots the system]  
[When the system reboots, log in]
```

```
$ DELETE DUA2:[SYSTEM] JUNK.SYS;
```

2.16 ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの作成と変更

性能を向上させるため、あるいはディスク空間の制約から、システム・ディスク以外のディスクにページ・ファイル, スワップ・ファイル, およびシステム・ダンプ・ファイルを作成する

ことがあります。ただし、エラー・ログ・ファイルは、システム・ディスク上になければなりません。

次の項では、ページ・ファイル、スワップ・ファイル、およびシステム・ダンプ・ファイルのさまざまな作成方法を説明します。

方法	参照箇所
AUTOGEN を使用 (標準的な方法)	2.16.1 項 「AUTOGEN を使用 (標準的な方法)」
SWAPFILES.COM を使用 (1 次ファイルのみ)	2.16.2 項 「SWAPFILES.COM の使用方法」
SYSGEN を使用	2.16.3 項 「SYSGEN の使用法」

2.16.1 AUTOGEN を使用 (標準的な方法)

新しく作成し AUTOGEN を起動するファイルの名前、格納場所、およびサイズを指定するように、MODPARAMS.DAT にシンボルを追加することにより、AUTOGEN を使用して新しいシステム・ページ・ファイル、スワップ・ファイル、およびダンプ・ファイルを作成することができます。これらの作業を行う前に、AUTOGEN およびパラメータ・ファイル MODPARAMS.DAT を理解しておく必要があります。AUTOGEN については 1.4 項 「AUTOGEN コマンド・プロシージャについて」、MODPARAMS.DAT については 1.4.4 項 「AUTOGEN パラメータ・ファイル (MODPARAMS.DAT) について」 をそれぞれ参照してください。

AUTOGEN はシステム・ページ・ファイル、スワップ・ファイル、およびダンプ・ファイルの適切なサイズを自動的に計算します。また、ファイルを適切なサイズに変更し、インストールします。AUTOGEN が計算するサイズは、ファイル MODPARAMS.DAT でシンボルを定義することにより制御できます。詳細は 2.16.1.2 項 「MODPARAMS.DAT でのページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズの制御」 を参照してください。

作業方法

ページ・ファイル、スワップ・ファイル、およびダンプ・ファイルのサイズを変更するには、AUTOGEN を 2 段階に分けて実行します。

- 次のコマンドを入力して、AUTOGEN の第 1 段階を起動する。AUTOGEN は、システム・ファイルのサイズの計算結果を SYS\$OUTPUT に表示する。

```
$ @SYS$UPDATE:AUTOGEN SAVPARAMS TESTFILES
```

- ステップ 1 で表示されたファイルのサイズが十分ではない場合は、2.16.1.2 項 「MODPARAMS.DAT でのページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズの制御」 で説明したように、ファイルのサイズを制御するために MODPARAMS.DAT にシンボルを追加し、ステップ 1 に戻る。
- ステップ 1 で表示されたファイルのサイズで十分な場合は、次のコマンドを使用して AUTOGEN の第 2 段階を実行する。これにより、システムがリブートする時に、変更されたシステム・ファイルをインストールされる。

```
$ @SYS$UPDATE:AUTOGEN GENPARAMS REBOOT
```

- システムがブートされるたびに、変更されたファイルがインストールされるように、サイト固有のスタートアップ・コマンド・プロシージャ SYPAGSWPFILES.COM にコマンドを追加する。詳細は 2.14 項 「ページ・ファイルとスワップ・ファイルのインストール」 参照。

2.16.1.1 システム・ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの格納場所の制御

作成するページ・ファイルとスワップ・ファイルの名前と格納場所を指定するには、次のシンボルを MODPARAMS.DAT に追加します。

定義	ページ・ファイル	スワップ・ファイル	ダンプ・ファイル
ファイル名と格納場所	PAGEFILE n _NAME = "ファイル指定"	SWAPFILE n _NAME = "ファイル指定"	DUMPFIL E _DEVICE= "デバイス"

ただし、

- n はページ・ファイルまたはスワップ・ファイルを指定する。1 次ページ・ファイルと 1 次スワップ・ファイルを指定するときは、 n に対して 1 を指定する。その後のファイルを指定するときは、 n に対して順に大きい整数値を指定する。たとえば、2 次ページ・ファイルまたは 2 次スワップ・ファイルを参照するときは、 n の値として 2 を指定する。
- **ファイル指定** は、作成するファイルの完全なファイル指定であり、二重引用符 ("") で囲む必要がある。
- **デバイス** は、使用するディスクの名前を指定し、二重引用符 ("") で囲む必要がある。

2.16.1.2 MODPARAMS.DAT でのページ・ファイル, スワップ・ファイル, ダンプ・ファイルのサイズの制御

AUTOGEN パラメータ・ファイル MODPARAMS.DAT に情報を追加すると、AUTOGEN が計算するページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズを制御することができます。MODPARAMS.DAT にシステム・ファイルのサイズ情報を指定しないと、AUTOGEN は、ページ・ファイル、スワップ・ファイル、ダンプ・ファイルの省略時のサイズ計算を行います。

次のいずれかを指定するシンボルを、MODPARAMS.DAT で定義できます。

指定するサイズ	参照箇所
すべてのページ・ファイルまたはスワップ・ファイルに必要なサイズの合計。(システム・ダンプ・ファイルに対しては無効)。	表 2-4 「ページ・ファイル、スワップ・ファイル、ダンプ・ファイル、エラー・ログ・ダンプ・ファイル空間の合計サイズを制御するシンボル」
ページ・ファイル、スワップ・ファイル、またはダンプ・ファイルの個々のサイズ。	表 2-5 「個々のページ・ファイルおよびスワップ・ファイルのサイズを制御するためのシンボル」



注意:

ファイルの合計サイズと個々のサイズの両方を指定することはできません。MODPARAMS.DAT 中のシンボル定義が矛盾する場合は、AUTOGEN は警告メッセージを表示します。

ページ・ファイルとスワップ・ファイルに関しては、他にページ・ファイルおよびスワップ・ファイルがない場合のみ、AUTOGEN は SYS\$SYSTEM:PAGEFILE.SYS および SYS\$SYSTEM:SWAPFILE.SYS を操作します。2 次ファイルがある場合は、AUTOGEN は 2 次ファイルを操作し、1 次ファイルは操作から除外されます。ただしインスタンスによっては、1 次ページ・ファイルおよび 1 次スワップ・ファイルが変更されることがあります。

VAX システムでは、システム・ダンプ・ファイルに対して、AUTOGEN は 1 つのファイルだけのサイズを操作します。DUMPFIL E _DEVICE が指定されていないときは、システム・ディスクのシステム・ダンプ・ファイルを操作し、DUMPFIL E _DEVICE が指定されているときは、指定されたデバイスのシステム・ダンプ・ファイルを操作します。

VAX システムでは、DUMPFIL E _DEVICE が指定されているときは、AUTOGEN はエラー・ログ・バッファに対して常に最小の SYS $DUMP$.DMP ファイルをシステム・ディスク上に作成します。

Alpha システムおよび I64 システムでは、AUTOGEN はシステム・ディスクのエラー・ログ・ダンプ・ファイルのサイズだけを操作します。

AUTOGEN が 1 次ファイルのサイズを変更することが望ましくない場合には、MODPARAMS.DAT に次のシンボルを指定してください。

```
PAGEFILE = 0
SWAPFILE = 0
DUMPFIL = 0
ERRORLOGDUMP = 0 ! Alpha and I64
```

これらのシンボルは、サイズを計算するときに、1 次ページ・ファイル、1 次スワップ・ファイル、および 1 次ダンプ・ファイルが無視するように AUTOGEN に要求します。

システム・ページ・ファイル、スワップ・ファイル、ダンプ・ファイルの作成または拡張によって、ターゲット・ディスクの 95 パーセント以上が使用されると、AUTOGEN は警告を出し操作を実行しません。

しかし、Alpha システムおよび I64 システムでは、95 パーセントという規則はエラー・ログ・ダンプ・ファイル、SYS\$ERRLOG.DMP に適用されません。ディスクにファイルを格納できる場合には、このファイルが作成されます。

AUTOGEN を使用すると、現在のページ・ファイル、スワップ・ファイル、ダンプ・ファイルよりも小さなファイルを作成できます。システムをブートして新しいファイルを使用したら、必ず DCL コマンドの PURGE を実行してください。

現在インストールされているページ・ファイルおよびスワップ・ファイルのサイズを確認するには、DCL コマンドの SHOW MEMORY/FILES を実行します。これらのファイルのサイズを増やしたもののリポートしていない場合は、変更前のサイズが表示されます。ダンプ・ファイルのサイズを確認するには、DIRECTORY コマンドを使用します。



注意:

値 0 を指定した場合、または現在のサイズの 10 パーセント以内のサイズを指定した場合、AUTOGEN はファイルのサイズを変更しません。

ページ・ファイル、スワップ・ファイル、システム・ダンプ・ファイル、またはエラー・ログ・ダンプ・ファイルの全部の領域を制御するために、MODPARAMS.DAT で定義できるシンボルの一覧を表 2-4 「ページ・ファイル、スワップ・ファイル、ダンプ・ファイル、エラー・ログ・ダンプ・ファイル空間の合計サイズを制御するシンボル」に示します。

表 2-4 ページ・ファイル、スワップ・ファイル、ダンプ・ファイル、エラー・ログ・ダンプ・ファイル空間の合計サイズを制御するシンボル

操作	ページ・ファイル・シンボル	スワップ・ファイル・シンボル	ダンプ・ファイル・シンボル	エラー・ログ・ファイル・シンボル
空間の総量の定義	PAGEFILE = n ¹	SWAPFILE = n ¹	DUMPFIL = n ¹	ERRORLOGDUMP = n ¹
合計サイズの拡大	ADD_PAGEFILE = n	ADD_SWAPFILE = n	ADD_DUMPFIL = n	ADD_ERRORLOGDUMP = n
合計サイズの最大値の指定	MAX_PAGEFILE = n	MAX_SWAPFILE = n	MAX_DUMPFIL = n	MAX_ERRORLOGDUMP = n
合計サイズの最小値の指定	MIN_PAGEFILE = n	MIN_SWAPFILE = n	MIN_DUMPFIL = n	MIN_ERRORLOGDUMP = n

¹ n はブロック単位の合計サイズ。n が 0 の場合、AUTOGEN の対応するセクションはスキップされる。ページ・ファイルとスワップ・ファイルに関しては、n が 0 でない場合、2 次ファイルが存在しなければ、AUTOGEN はその値を 1 次ファイルに適用する。n が 0 でない場合、2 次ファイルが存在すれば、AUTOGEN は変更値をページ・ファイルまたはスワップ・ファイルのすべての 2 次ファイルに等分し、ほとんどの場合、1 次ファイルは変更しない。

表 2-5 「個々のページ・ファイルおよびスワップ・ファイルのサイズを制御するためのシンボル」は、個々のファイルのサイズを制御する場合に MODPARAMS.DAT に定義できるシンボルの一覧です。

表 2-5 個々のページ・ファイルおよびスワップ・ファイルのサイズを制御するためのシンボル

操作	ページ・ファイル・シンボル ¹	スワップ・ファイル・シンボル ¹
ファイル・サイズの指定	PAGEFILE _n _SIZE = ブロック・サイズ	SWAPFILE _n _SIZE = ブロック・サイズ
ファイル・サイズの拡大	ADD_PAGEFILE _n _SIZE = ブロック・サイズ	ADD_SWAPFILE _n _SIZE = ブロック・サイズ
最大ファイル・サイズの指定	MAX_PAGEFILE _n _SIZE = ブロック・サイズ	MAX_SWAPFILE _n _SIZE = ブロック・サイズ
最小ファイル・サイズの指定	MIN_PAGEFILE _n _SIZE = ブロック・サイズ	MIN_SWAPFILE _n _SIZE = ブロック・サイズ

1 **n** には、ページ・ファイルまたはスワップ・ファイルを示す整数を指定する。1 は 1 次ページ・ファイルまたは 1 次スワップ・ファイルを示し、それ以降のファイルでは値が増えていく。たとえば、2 次ページ・ファイルまたは 2 次スワップ・ファイルを指定する場合には、**n** の値を 2 にする。ブロック・サイズにはサイズをブロック単位で指定する。

例

- すべてのページ・ファイルの合計サイズを 100,000 ブロック以下にするには、次の行を MODPARAMS.DAT に追加します。

```
PAGEFILE = 100000
```

1 次ページ・ファイルだけしか存在しない場合、そのファイルのサイズは、100,000 ブロックになります。ページ・ファイルが複数存在する場合には、現在の合計サイズと新しい合計サイズとの差が 2 次ファイルに配分されます。たとえば、PAGEFILE = 100000 を指定すると、変更後のページ・サイズは次のようになります。

ファイル	元のサイズ(ブロック数)	変更後のサイズ(ブロック数)
1 次ページ・ファイル	10,000	10,000
2 次ページ・ファイル 1	30,000	45,000
2 次ページ・ファイル 2	30,000	45,000

- 1 次ページ・ファイルのサイズを 10,000 ブロックに設定するよう、AUTOGEN に指示するためには、シンボルを次のように定義します。

```
PAGEFILE1_SIZE = 10000
```

- サイズが 30,000 ブロックの新しい 2 次スワップ・ファイル PAGED\$:[PAGESWAP]SWAPFILE.SYS を作成するよう、AUTOGEN に指示するためには、シンボルを次のように定義します。

```
SWAPFILE2_NAME = "PAGED$:[PAGESWAP]SWAPFILE.SYS"
MIN_SWAPFILE2_SIZE = 30000
```

2.16.2 SWAPFILES.COM の使用方法

ページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズを変更する場合は、できるだけ AUTOGEN を使用するようにしてください。ただし、1 次ページ・ファイル、1 次スワップ・ファイル、1 次ダンプ・ファイルのサイズは、コマンド・プロシージャ SYS\$UPDATE:SWAPFILES.COM を使用して変更することができます。SWAPFILES.COM によ

り、サイズを変更する前に、ページ・ファイル、スワップ・ファイル、ダンプ・ファイルの現在のサイズを見ることができます。

ページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズを変更する場合は、2.16.1.2 項「MODPARAMS.DAT でのページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズの制御」で説明したように、MODPARAMS.DAT を編集して新しいサイズを指定する必要があります。新しいサイズを MODPARAMS.DAT に指定しなかった場合、AUTOGEN は次回実行されたときにファイルのサイズを変更します。

このプロシージャにより、SYS\$SYSTEM 内の現在のページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズと、システム・ディスクに残っている空間の量が表示され、新しいサイズを入力したり、既存のサイズを保持したりすることができるようになります。既存のファイル・サイズよりも大きいサイズを指定すると、このプロシージャにより、ページ・ファイルまたはダンプ・ファイルのサイズが自動的に増やされます。また、システムのページ・ファイル、スワップ・ファイル、ダンプ・ファイルに、小さいサイズを指定すると、新しくファイルが作成されます。

作業方法

1. 次のコマンドを入力して、コマンド・プロシージャを起動する。

```
$ @SYS$UPDATE:SWAPFILES.COM
```

システムは SYS\$SYSTEM 内にある現在のファイルおよびそのサイズを表示する。次の例を参照。

```
Current file sizes are:
```

```
Directory SYS$SYSROOT:[SYSEXE]
```

```
PAGEFILE.SYS;1      16384
SYSDUMP.DMP;1       4128
SWAPFILE.SYS;1      3072
```

```
Total of 3 files, 23584 blocks.
```

```
There are 128741 available blocks on SYS$SYSDEVICE.
```

2. 次のプロンプトに対して、ページ・ファイルに必要なサイズをブロック単位で入力する。サイズを変更しない場合は Return を押す。

```
Enter new size for page file:
```

3. 次のプロンプトに対して、ダンプ・ファイルに必要なサイズをブロック単位で入力する。サイズを変更しない場合は Return を押す。

```
Enter new size for system dump file:
```

4. 次のプロンプトに対して、スワップ・ファイルに必要なサイズをブロック単位で入力する。サイズを変更しない場合は Return を押す。

```
Enter new size for swap file:
```

5. システムをシャットダウンしてからリブートし、新しいファイルを使用できるようにする。
6. システムをリブートした後、使用しなくなったファイルのコピーをパージする。システムをリブートするまでは、古いファイルを削除しないこと。
7. MODPARAMS.DAT を編集して、新しいファイル・サイズを取り込む (2.16.1.2 項「MODPARAMS.DAT でのページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズの制御」参照)。MODPARAMS.DAT に新しいサイズを指定していない場合は、AUTOGEN が次回実行されるとき、自動的にファイルのサイズを変更する。

例

```
$ @SYS$UPDATE:SWAPFILES
```

```
To leave a file size at its current value type a carriage return in response to its size prompt. Current file sizes are:
```

```
Directory SYS$SYSROOT:[SYSEXE]
```

```
PAGEFILE.SYS;1      100000
SYSDUMP.DMP;1       28000
SWAPFILE.SYS;1      33000
```

```
Total of 3 files, 161000 blocks.
```

```
There are 128741 available blocks on SYS$SYSDEVICE.
```

```
Enter new size for page file: Return
```

```
Enter new size for system dump file: 30000
```

```
%SYSGEN-I-EXTENDED, SYS$SYSROOT:[SYSEXE]SYSDUMP.DMP;1 extended
```

```
Enter new size for swap file: Return
```

```
*****
* Please reboot in order for the new files to be used by the system. *
* After rebooting, purge obsolete copies of the files.                *
* DO NOT delete the old files until after the reboot.                  *
*****
```

2.16.3 SYSGEN の使用法

ページ・ファイル、スワップ・ファイル、ダンプ・ファイルを作成する場合には、できるだけ AUTOGEN を使用してください。AUTOGEN は、システム生成ユーティリティ (SYSGEN) を起動して、ファイルを作成または変更します。ただし、緊急時には、システム生成ユーティリティ (SYSGEN) を使用して、ページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズを直接変更することができます。たとえば、ページ・ファイルの空間が危険なほど少なくなっていることが分かった場合には、SYSGEN を利用して、ページ・ファイルの空間をすぐに追加し、システムのハングを防ぐことができます。



注意:

VAX:

システム・パラメータ SWPFILCNT および PAGFILCNT はそれぞれ、システムがインストールするスワップ・ファイルおよびページファイルの数を制限します。詳細は『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

Alpha および I64:

OpenVMS Alpha および I64 は、1 つのシステムで最大で 254 のページ・ファイルまたはスワップ・ファイルあるいはその両方をサポートします。システム・パラメータ SWPFILCNT および PAGFILCNT は、Alpha システムおよび I64 システムでは使用されません。

作業方法

1. ファイルの位置と適切なサイズを調べる (2.4 項「ダンプ・ファイル、ページ・ファイル、スワップ・ファイルのサイズの机上計算」を参照)。
2. SYSGEN を起動し、次の形式で CREATE コマンドを入力する。
CREATE ファイル指定/SIZE= ブロック・サイズ
ファイル指定には、完全ファイル指定を行う。
ブロック・サイズには、ファイルのサイズをブロック数で指定する。

指定したファイルがすでに存在する場合、指定したサイズが既存のファイルのサイズより大きければ、既存のファイルが拡大される。指定したファイルがすでに存在する場合、指定したサイズが既存のサイズより小さければ、指定したサイズのファイルが新しく作成される。

たとえば、次のコマンドは既存のページ・ファイル PAGEFILE.SYS を拡大する。

```
SYSGEN> CREATE PAGEFILE.SYS/SIZE=100000
```

SYSGEN の CREATE コマンドについての詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』の SYSGEN の節を参照。



注意:

ファイルの作成および削除を頻繁に行うと、ディスク上の空き空間が著しくフラグメンテーションする場合があります。システム・ファイルの作成または拡張によって、そのファイルがフラグメンテーションされ、システムがブートできなくなると判断した場合、SYSGEN は HEADERFULL 警告メッセージを出します。そのような場合には、ボリューム上の空き空間を1つの連続領域にまとめるため、システム・ボリュームをいったんバックアップしてから復元し (『OpenVMS システム管理者マニュアル (上巻)』参照) します。

その後、SYSGEN の操作を再試行します。SYSGEN が警告メッセージを出す場合には、ファイルのサイズは多少大きくなりますが、CREATE コマンドで指定したサイズまでにはなりません。

3. 新しいファイルまたは変更したファイルを使用するために、リブートする必要があるかどうかを判断する。次の表を参照。

種類	変更形態	リブート
1 次ページ・ファイル 1 次スワップ・ファイル 1 次システム・ダンプ・ファイル エラー・ログ・ダンプ・ファイル ¹	作成	必要
	拡大	必要
2 次ページ・ファイル 2 次スワップ・ファイル	作成	不要 ²
	拡大	必要
代替 (DOSD) ダンプ・ファイル (Alpha および I64)	作成	不要
	拡大	不要
代替 (DOSD) ダンプ・ファイル (VAX)	作成	必要
	拡大	必要

- 1 1 次ページ・ファイル、スワップ・ファイル、ダンプ・ファイルは、それぞれ SYS\$SPECIFIC:[SYSEXE]PAGEFILE.SYS、SWAPFILE.SYS、SYSDUMP.DMP。エラー・ログ・ダンプ・ファイルは SYS\$ERRLOG.DMP。
- 2 システムをリブートする必要はないが、システムで 2 次ファイルを利用する前に 2 次ファイルをインストールしなければならない。詳細は 2.14 項「ページ・ファイルとスワップ・ファイルのインストール」を参照。

4. ファイルの新しいバージョンを作成した場合には、システムのリブート後、古いバージョンをパージする。
5. システムがブートされるたびに、ファイルが確実にインストールされるように、サイト固有のスタートアップ・コマンド・プロシージャ SYPAGSWPFILES.COM にコマンドを追加する。操作手順については、2.14 項「ページ・ファイルとスワップ・ファイルのインストール」を参照。

6. AUTOGEN が計算に従ってファイルのサイズを変更することが望ましくない場合には、MODPARAMS.DAT を変更して、これらのファイルのサイズを指定する。2.16.1.2 項「MODPARAMS.DAT でのページ・ファイル、スワップ・ファイル、ダンプ・ファイルのサイズの制御」を参照。

例

既存のファイル PAGEFILE.SYS, SWAPFILE.SYS, および SYSDUMP.DMP を指定したサイズに拡大します。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> CREATE PAGEFILE.SYS/SIZE=100000
%SYSGEN-I-EXTENDED, SYS$SYSROOT:[SYSEXE] PAGEFILE.SYS;1 extended
SYSGEN> CREATE SWAPFILE.SYS/SIZE=30000
%SYSGEN-I-EXTENDED, SYS$SYSROOT:[SYSEXE] SWAPFILE.SYS;1 extended
SYSGEN> CREATE SYSDUMP.DMP/SIZE=33000
%SYSGEN-I-EXTENDED, SYS$SYSROOT:[SYSEXE] SYSDUMP.DMP;1 extended
SYSGEN> EXIT
```

2.17 プロセス・ダンプについて

単一プロセスが異常終了しても、オペレーティング・システムがまだ実行されている場合には、システムは、プロセスが異常終了した原因を判断するための補助として、プロセスに関する情報が含まれている**プロセス・ダンプ**を作成することができます。

省略時の設定では、プロセス・ダンプは現在の省略時のディレクトリに書き込まれます。これは、代替ディレクトリ・パスを識別するように論理名 SYS\$PROCDDMP を定義することによって、上書きすることができます。プロセス・ダンプ・ファイルの名前は、プロセス・ダンプが書き込まれる時点でアクティブなメイン・イメージの名前と常に同じで、ファイル・タイプが .DMP であることに注意してください。

Alpha システムおよび I64 システムでは、プロセス・ダンプには完全プロセス・ダンプと部分プロセス・ダンプがあります。**完全プロセス・ダンプ**には、プロセス・スペースのすべてと、システム・スペースからの、プロセスに関連するデータのすべてが含まれます。**部分プロセス・ダンプ**には、プロセス・スペースからは、ユーザが読み込み可能なデータのみ、システム・スペースからはデータ構造のうち、機密度が高いと見なされないものだけが含まれます。他社製ソフトウェアの暗号キーなど、特権が与えられたり保護されているデータは、機密度が高いとみなされます。

Alpha システムおよび I64 システムでは、DCL コマンド SET PROCESS/DUMP=NOW process-spec を使用して、別のプロセスのダンプを強制的に書き込むことができます。このコマンドを実行すると、process-spec によって占有されているアドレス空間の内容が、process-spec の現在のディレクトリの image-name.DMP というファイルに直ちに書き込まれます。

DCL コマンド SET PROCESS/DUMP の詳細については、『OpenVMS DCL デクシヨナリ: N-Z』を参照してください。

2.17.1 特権ユーザおよびプロセス・ダンプへのアクセスについて (Alpha および I64)

ここでは、特権ユーザとは、次のいずれかの条件に一致するユーザです。

- CMKRNL, CMEXEC, SYSPRV, READALL, または BYPASS のうち 1 つまたは複数の特権を持つ。
- システム UIC グループ (省略時の設定では [10,n] 以下) のメンバである。このようなユーザは、SYSPRV 特権を持っているかのように扱われる。

CMKRNL または CMEXEC を持っていれば、完全プロセス・ダンプを書き込むことができます。ここに示された他の特権を持っていれば、プロセス・ダンプがどこに書き込まれていても、これを読み込むことができます。

通常、非特権ユーザは完全プロセス・ダンプを読み取ることができてはならないため、省略時の設定ではこれを読み取ることができません。ただし、状況によっては、非特権ユーザが完全プロセス・ダンプを読み取ることが必要な場合もあります。また、非特権ユーザが完全プロセス・ダンプを作成できる一方で、部分プロセス・ダンプだけを読み込むことができるようにすることが必要な場合もあります。

ライト識別子 `IMGDMP$READALL` により、非特権ユーザも完全プロセス・ダンプを読み込むことができるようになります。ライト識別子 `IMGDMP$PROTECT` により、完全プロセス・ダンプが、そのプロセス・ダンプを作成した非特権ユーザによって読み込まれるのを防ぐことができます。これらのライト識別子は、OpenVMS のインストール中にイメージ `SYS$SYSTEM:IMGDMP_RIGHTS.EXE` によって作成されます。このイメージは、システム・スタートアップ中にも自動的に実行され、これらのライト識別子が確実に正しい値および属性を伴って存在するようにします。

これらのライト識別子が削除されている場合には、`SYS$SYSTEM:IMGDMP_RIGHTS.EXE` を実行すると再作成することができます。次に例を示します。

```
$ RUN SYS$SYSTEM:IMGDMP_RIGHTS
%PROCDUMP-I-CREATED, rights identifier IMGDMP$READALL successfully created
%PROCDUMP-I-CREATED, rights identifier IMGDMP$PROTECT successfully created
IMGDMP$READALL には属性がありませんが、IMGDMP$PROTECT は RESOURCE 属性を伴って作成されることに注意してください。
```

2.17.2 プロセス・ダンプへのアクセスの許可 (Alpha および I64)

非特権ユーザが完全プロセス・ダンプの書き込み/読み込みができるようにするには、そのユーザにライト識別子 `IMGDMP$READALL` を許可します。`IMGDMP$READALL` ライト識別子がない場合には、イメージ `SYS$SYSTEM:IMGDMP_RIGHTS.EXE` を実行してこれを作成します (2.17.1 項「特権ユーザおよびプロセス・ダンプへのアクセスについて (Alpha および I64)」を参照)。次に `AUTHORIZE` を使用して、このライト識別子をユーザに許可します。次に例を示します。

```
$ DEFINE /USER SYSUAF SYS$SYSTEM:SYSUAF.DAT !if necessary
$ RUN SYS$SYSTEM:AUTHORIZE
UAF> GRANT /IDENTIFIER IMGDMP$READALL <user>
UAF> EXIT
```

ライト識別子を実行できるようにするには、ユーザはいったんログアウトし、再びログインしなければなりません。ライト識別子 `IMGDMP$READALL` を持つ非特権ユーザは、無制限に完全プロセス・ダンプの読み込み/書き込みができます。

2.17.3 プロセス・ダンプへのアクセスの制限 (Alpha および I64)

非特権ユーザが完全プロセスを書き込むことができるようにしながら、同時にユーザがそのプロセス・ダンプを読み込むことができないようにすることができます。その場合、次の手順を行います。

1. `IMGDMP$PROTECT` ライト識別子が存在しない場合には、イメージ `SYS$SYSTEM:IMGDMP_RIGHTS.EXE` を実行してこれを作成する (2.17.1 項「特権ユーザおよびプロセス・ダンプへのアクセスについて (Alpha および I64)」を参照)。
2. ライト識別子 `IMGDMP$PROTECT` を使用して、保護されたディレクトリを作成する。次に例を示す。

```
$ CREATE /DIRECTORY DKA300:[PROCDUMPS] -
    /PROTECTION=(S:RWE,O:RWE,G,W) /OWNER_UIC=IMGDMP$PROTECT
$ SET SECURITY DKA300:[000000]PROCDUMPS.DIR -
    /ACL=( (DEFAULT_PROTECTION,SYSTEM:RWED,OWNER:RWED,GROUP:,WORLD:), -
    (IDENTIFIER=IMGDMP$PROTECT,ACCESS=READ+WRITE), -
    (IDENTIFIER=IMGDMP$PROTECT,OPTIONS=DEFAULT, -
    ACCESS=READ+WRITE+EXECUTE+DELETE+CONTROL), -
    (CREATOR,ACCESS=NONE) )
```

3. エグゼクティブ・モード論理名 `SYS$PROTECTED_PROCDMP` が、保護されたディレクトリを指すように定義する。次に例を示す。

```
$ DEFINE /SYSTEM /EXECUTIVE_MODE SYS$PROTECTED_PROCDMP DKA300:[PROCDUMPS]
```

4. 保護されたディレクトリが含まれるディスクで `DISKQUOTA` を使用する場合には、プロセス・ダンプに使用する最大ディスク領域を指定する。次に例を示す。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> DISKQUOTA CREATE /DEVICE=DKA300 ! if necessary
SYSMAN> DISKQUOTA ENABLE /DEVICE=DKA300 ! if necessary
SYSMAN> DISKQUOTA ADD IMGDMPS$PROTECT /DEVICE=DKA300 /PERMQUOTA=10000
SYSMAN> DISKQUOTA REBUILD /DEVICE=DKA300 ! if necessary
SYSMAN> EXIT
```



警告:

`IMGDMPS$PROTECT` は、どのユーザにも許可しないでください。これは、プロセス・ダンプの書き込み中に、必要に応じてエグゼクティブ・モードから `SYS$SHARE:IMGDMPS.EXE` によって許可されたり、許可を取り消されたりします。これをユーザに永久に許可すると、ユーザは、保護されたディレクトリに書き込まれたすべてのプロセス・ダンプのすべてにアクセスできるようになります。

保護されたディレクトリにさらに `ACL` を追加して設定し、どのユーザにそのディレクトリ内のプロセス・ダンプへの読み込みや書き込みを許可するかを、さらに厳しく制御することもできます。

高い特権を使用してイメージがインストールされたり、保護されたサブシステムに所属する場合にプロセス・ダンプを取るには、ユーザは `CMKRNL` 特権を保持していなければならない、定義上は特権ユーザに限られます (2.17.1 項「特権ユーザおよびプロセス・ダンプへのアクセスについて (Alpha および I64)」を参照)。

第3章 性能の管理

この章では、性能管理の基本概念を紹介します。詳細は『Guide to OpenVMS Performance Management』を参照してください。

この章の内容

この章では、次の作業を説明します。

作業	参照箇所
作業負荷の把握	3.2 項 「作業負荷の把握」
作業負荷管理方針の選択	3.3 項 「作業負荷管理方針の決定」
作業負荷の配分	3.4 項 「作業負荷の配分」
チューニングが必要な時期の予測	3.6 項 「チューニングが必要な時期の予測」
チューニングの評価	3.7 項 「チューニング結果の評価」
性能オプションの選択	3.8 項 「性能オプションの選択」
インストール・ユーティリティによるイメージのインストール (INSTALL)	3.10 項 「INSTALL による既知イメージのインストール」
特定用途のためのメモリの予約 (Alpha および I64)	3.11 項 「予約メモリ・レジストリ」

さらに、次の項目について説明します。

項目	参照箇所
性能管理	3.1 項 「性能管理について」
システム・チューニング	3.5 項 「システム・チューニングについて」
イメージと既知イメージ	3.10.1 項 「イメージおよび既知イメージについて」
既知のファイル・リスト	3.10.2 項 「既知ファイルのエントリについて」
既知イメージの属性	3.10.3 項 「既知イメージに割り当てることができる属性」

3.1 性能管理について

性能管理とは、現在の作業負荷のハードウェアとソフトウェアの資源を最適化することです。この仕事には、いくつかの異なる、しかし互いに関連する作業が伴います。

- システムの作業負荷とシステム資源の使用状況を把握する。それと同時に OpenVMS が資源をどのように管理しているかを理解すれば、次の項目の現実的な標準値を確立することができる。
 - 会話型処理およびバッチ処理におけるスループット
 - 会話型処理における応答時間
 - バッチ・ジョブのターンアラウンド時間
- システムの動作を定期的に監視して、各資源がいつ、どのようなときに限界に近づくかを調べる。
- ユーザから報告される性能低下の事例を調査する。
- システムの作業負荷配分またはハードウェア構成の変更計画を作成し、システム・パラメータ値を調整するための準備を行う。
- インストール後、オプションのシステム管理処理を行う。

3.2 作業負荷の把握

システムの正常時の作業負荷と動作を把握することは、システム管理者が性能を評価するときに最も重要なことの1つです。各システム管理者は、システムの作業負荷を十分に把握し、次の項目を確認してください。

- 各時間帯における典型的なユーザ数
- 各時間帯における、典型的なユーザ数のもとでのさまざまなタスクの典型的な応答時間
- 動作がピークになる時間帯
- 一般に、どの時刻にどのジョブが実行されるか？
- よく実行されるジョブのうち、CPU、メモリ、ディスク空間を集中して消費するものはどれか？
- イメージを最も頻繁に起動するアプリケーションはどれか？
- デバイス・ドライバなどのシステム・ソフトウェアに変更されたり、ユーザによって追加された部分があるとしたら、それはどこか？
- システムにボトルネックは存在していないか？今後発生が予想されるボトルネックはないか？

OpenVMS オペレーティング・システムの管理が初めての方は、システム動作を監視する際に次のツールを利用するといいでしょう。

- Monitor ユーティリティ
- 会計情報ユーティリティ
- SHOW コマンド (DCL で利用可)

『Guide to OpenVMS Performance Management』(VAX システムの場合)、および『Guide to OpenVMS AXP Performance Management』(Alpha システムの場合)には、Monitor ユーティリティなどの OpenVMS ツールを使ったシステムの性能の監視と評価の手順が詳しく説明されています。また、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』には、Monitor ユーティリティの使用に関する参照情報が記述されています。

時間の経過とともに、システムの典型的なページ・フォルト率、典型的な CPU 使用量、正常時のメモリ使用量、典型的な動作モードといった、性能を示す基準値が把握できます。また、特定の処理がシステムの性能にどのように影響するか、ユーザ数や時刻によって、システムの性能のどの部分にどのような影響が出るかが分かり始めます。

システムの監視を続けるうちに、許容できる値の範囲を理解できるようになり、同じツールでより効果的に異常を検出できるようになります。性能の管理を効果的に行うためには、システムを定期的に評価することが重要です。問題を避ける最良の方法は、それを予測することです。問題が大きくなるうちに、システムがどのように動作しているのかを把握するように努めてください。



注意:

Monitor ユーティリティと会計情報ユーティリティを使って重要なデータ項目を定期的に分析すれば、システムの動作をより理解できます。また、これらのデータの収集と観察を続けることによって、使用量の傾向が分かり、システムがいつその限界に達するかを予測することができます。

さらに、システム管理用のツールによってもシステム資源が使用されるということを知っておきましょう。計測する項目やデータ収集の頻度を決めるときには、そのことに十分注意してください。ツールを過度に使用すると、データの収集、保存、分析によって資源が使用されるために、システムの作業負荷と容量を正しく認識できなくなることがあります。データの収集と分析は、事前に計画をたてた上で、その計画に従って実行するようにしてください。

3.3 作業負荷管理方針の決定

システムの性能は作業負荷管理の効率に比例して上下します。作業負荷を管理するための方針は、各システムで独自に決定する必要があります。システムの設定値を調整する前に、次の点を必ず解決してください。

- 一日のうちで作業負荷が「ピークになる」時間帯、すなわち他の時間帯より著しく負荷が大きくなる時間帯はあるか？
- 作業負荷のバランスを良くする方法はあるか？
負荷のバランスを少しでも改善できるように各ユーザに協力してもらうのも1つの方法である。
- 負荷の小さい時間帯に、バッチ形式で実行できるジョブはあるか？
- 主操作時間と副操作時間を各ユーザが採用しているか？採用していない場合、その方法を採用することによってシステムの性能は向上しないか？採用している場合、選択されている時間帯はすべてのユーザに対して最適か？
ユーザおよびアプリケーションを追加あるいは削除したときには、このことを必ず検討し、適切なバランスを保つ必要がある。
- 今後追加するアプリケーションは、システムに現存するボトルネックあるいは今後予想されるボトルネックを回避できるように設定できるか？同じ目的で、現在のアプリケーションに対して何らかの設定変更ができるか？『Guide to OpenVMS File Applications』を参照。
- オペレーティング・システムのコード共有機能を最大限活用しているか？
コード共有機能を利用すれば、メモリの使用量を抑えて、システムの性能を向上させることができる。

3.4 作業負荷の配分

システムの稼働時間全体に渡って、作業負荷をできるだけ均等に配分してください。ただし、サイト別の作業スケジュールをたてると、会話型ユーザを最適な時間に割り当てることがむずかしくなる場合もあります。次の方法を参考にしてください。

- 大型のジョブはバッチ・ジョブとして実行する。
大型のジョブはできるかぎりバッチにしてキューに登録する、という方針をたてる。バッチ・ストリームの数を制御して、会話形式のジョブが少ないときにバッチ型のジョブが多くなるようにする。また、DCL コマンドの修飾子を利用して、バッチ・ジョブを低い優先順位で実行したり、ワーキング・セットのサイズを調整したり、同時に実行できるジョブの数を制御したりする方法もある。バッチ環境の設定方法については、『OpenVMS システム管理者マニュアル (上巻)』を参照。
- システムの使用量を制限する。
一度にログインできるユーザの数を、システムが適切な応答時間でサポートできる数に制限する。会話型ユーザの数は、DCL の SET LOGINS/INTERACTIVE コマンドによって制限できる。また、同時に実行できるプロセスの数は、システム・パラメータ MAXPROCESSCNT を使って制御できる。システムに同時にアクセスできる遠隔ターミナルの数は、システム・パラメータ RJOBIM を使って制御できる。システム・パラメータの変更方法については、1.5 項「AUTOGEN によるシステム・パラメータの変更」を参照。『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』では、すべてのシステム・パラメータについて解説している。
さらに、ユーザをグループに分け、各グループがシステムを使用する日あるいは時間帯を制限することもできる。AUTHORIZE ユーティリティを使用すれば、各ユーザがログインできる時間帯を定義できる。具体的には、AUTHORIZE の修飾子 /PRIMEDAYS を参照。詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の AUTHORIZE の節を参照。

DCLのSET DAY コマンドによって、主曜日および副曜日の設定を変更できる。たとえば、主曜日が休日のときはそれを副曜日として指定する必要がある。

- 資源をバインドする必要性を減らすように各アプリケーションを設計する。
システムのボトルネックが現在どこにあるか、あるいは近い将来どこに発生するかを把握し、ボトルネックとなる資源の必要量を最小限に抑えるようにアプリケーションの導入計画を立てれば、作業負荷をより均等に配分することができる。『Guide to OpenVMS File Applications』を参照。

3.5 システム・チューニングについて

チューニングとは、さまざまなシステム設定値を変更することによって、特定の構成および作業負荷からシステム全体としての性能を向上させることです。メモリやデバイスを適切な時期に増設すれば、ほぼ確実にシステムの性能を大幅に向上させることができます。しかし、そのようなシステム構成の変更は、チューニングとは異なります。

ほとんどのシステムでは、作業負荷は常に変化しています。ある瞬間には最高の性能が得られても、次の瞬間に作業負荷が変化し、同じシステム・パラメータの設定でも、最高の性能が得られなくなることがあります。チューニングの最終目標は、システム全体としてベストな性能が平均的に得られる値を見つけることです。

性能に影響する問題の中には、次に示すように、システム設定値を調整しても解決できないものもあります。作業を始める前に、これらの問題が実際にはないかどうか調べる必要があります。

- 操作が適切でない。
- 到底達成し得ない性能を期待している。
- 使用しようとするアプリケーションに対してメモリが十分でない。
- 作業負荷に対してハードウェア構成が不十分である。たとえば、プロセッサ速度が遅い、デバイスのバスが少ない、ディスクの数が少ないなど。
- 作業負荷に対して適切なデバイスが選択されていない。たとえば、速度および容量が不十分であるなど。
- ハードウェアが正常に動作しない。
- アプリケーションの設計が不十分である。
- 1つのプロセスに対して、使用できるすべての資源の使用を許可している。

チューニングを行う場合、動作を注意深く分析し、値を変更する項目の数をできるだけ少なくします。システム資源は、次の2つのタイプのパラメータの値を調整することにより制御します。

パラメータ・タイプ	説明
システム・パラメータ	<p>システム・パラメータに設定した値により、システム全体としてのシステム資源が制御される。AUTOGEN コマンド・プロシージャは、システム構成のデータに従ってシステム・パラメータ値を自動的に設定する。さらに、稼働中のシステムからのフィードバックを記録し、システムの作業負荷に基づいて、パラメータ値を調整することができる。必要な調整を行うためのパラメータと新しい値の選択方法については、『Guide to OpenVMS Performance Management』(VAX システムの場合)、および『Guide to OpenVMS AXP Performance Management』(Alpha システムの場合)を参照。</p> <p>『Guide to OpenVMS Performance Management』では、必要な変更を行うためのパラメータと新しい値の選択方法について説明します。</p> <p>AUTOGEN を使用して、システム・パラメータ値を変更する方法については、1.5 項「AUTOGEN によるシステム・パラメータの変更」を参照。</p>
UAF 制限とクォータ	<p>利用者登録ファイル (UAF) レコードの制限とクォータに設定されている値により、ユーザごとのシステム資源が制御される。これらの値を制御するには、AUTHORIZE ユーティリティを使用する。詳細は『OpenVMS システム管理者マニュアル (上巻)』を参照。</p>

チューニングを始める前に『Guide to OpenVMS Performance Management』(VAX システムの場合)、または『Guide to OpenVMS AXP Performance Management』(Alpha システムの場合)を参照し、OpenVMS がどのように資源を管理しているか理解してください。また、各システム値の性格も理解しておく必要があります。これらが十分理解されていないと、結果としてシステム全体の性能を大幅に低下させてしまうことがあります。

3.6 チューニングが必要な時期の予測

OpenVMS システムでチューニングが必要になることはほとんどありません。AUTOGEN コマンド・プロシージャが、システム構成に依存するすべてのパラメータを実際の構成に適合するように自動的に設定するからです。AUTOGEN についての詳細は、1.4 項「AUTOGEN コマンド・プロシージャについて」を参照してください。

また、限定されますが、システムにはシステムの動作中にシステム自身を動的に調整する機能が備えられています。非ページング動的プール、ワーキング・セットのサイズ、空きページ・リスト、および変更ページ・リスト上のページの数といった特定の領域に関して、必要に応じて自動的に調整を行います。したがって、これらの値はシステムの動作中に動的に変化することがあります。

システムの性能が満足できない場合、その一般的な原因はハードウェアの容量不足です。システムに対する要求がその能力を超えてしまうと、システム設定値を調整しても性能が大幅に向上することはありません。そのような調整は、既存の資源のトレード・オフすなわち一時しのぎにすぎないからです。

しかし、チューニングの必要が全く無いわけではなく、特に次の条件を満たす場合にはチューニングが必要になることがあります。

- 現状の資源で最適な性能を得るよう、すでにチューニングを済ませている環境に、ディスクまたはメモリを増設する場合。この場合、新しい構成に合わせてチューニングをしない必要がある。まず最初に AUTOGEN コマンド・プロシージャを実行することが重要である。
- 作業負荷が大幅に変化すると予想される場合。新しい作業負荷に合わせてチューニングをする必要がある。

3.7 チューニング結果の評価

システムのチューニングを実施した場合、その後のシステム動作を監視して、期待した結果が得られたかどうかを調べてください。MONITOR ユーティリティおよび DCL の SHOW コマンドを利用します。SHOW コマンドについての詳細は、『OpenVMS DCL ディクショナリ』を参照してください。MONITOR コマンドの使用法については、6.7.2 項「MONITOR の起動」を参照してください。また、MONITOR コマンドについての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

たとえば、何度実行しても結果が同じであると考えられるいくつかのプログラムを、通常の作業負荷状況で実行します。チューニングの前後にほぼ同じ作業負荷のもとでこのプログラムを実行してその動作時間を計測すれば、比較のための基準が分かります。

しかし、この方法では作業負荷がほぼ等しい条件のもとで計測をしないと意味がありません。また、このテストだけではチューニングの成否を最終的に判断することはできません。調整した項目が計測対象のイメージだけに好結果をもたらし、他の部分には悪影響を与えている可能性もあるからです。したがって、どのような場合でもシステムの変更後しばらくは、システムの動作をよく観察してください。

3.8 性能オプションの選択

次に、オプションとして選択することができるシステム管理操作を示します。通常、これらの操作はインストール後に行われ、その結果、多くの場合には全体的な性能が向上します。実際

のシステム環境に合ったオプションを選択してください。すべてのオプションがどの環境にも適しているとはいえません。

- システム・ライブラリの展開。

オペレーティング・システムに付属する大規模なライブラリのほとんどは、ディスク使用量を抑えるために縮小 (圧縮) されている。ファイルが展開 (復元) されない限り、システムはアクセスのたびにファイルを動的に展開する必要がある。その結果、リンク動作やオンライン・ヘルプを呼び出すときに、特に性能の低下が目立つ。ディスク容量に余裕がある場合には、ライブラリをあらかじめ展開しておくこと、CPU パワーおよび応答時間の両方が向上する。システム・ライブラリの展開と LIBDECOMP.COM コマンド・プロシージャの使用法についての詳細は、3.9 項「システム・ライブラリの展開」を参照。

- ファイル・システムのハイウォーター・マークの無効化。

ハイウォーター・マークとは、ボリュームの初期化時、各ユーザが自分で書き込んでいないデータの読み込みを不可能にする機密保護機能で、省略時の値として設定されている。

非共用順次ファイルの場合、ハイウォーター・マークが性能に与える影響は最小限に抑えられる。ただし、非順次形式のファイルの場合、ハイウォーター・マーク処理により、オーバーヘッドがかかる。ファイルが作成されたり拡張されたりするたびに、割り当てられるディスク・ブロックの以前の内容が削除される。

この機能を無効にすることでシステムの性能がどのくらい向上するかは、次の要因によって決まる。

- 新規にファイルを作成する頻度
- 索引ファイルまたは相対ファイルの場合、既存のファイルを拡張する頻度
- ボリュームのフラグメンテーションの状態



注記: ハイウォーター・マーク機能を無効化する場合は、その前に機密保護上の問題がないかどうか考慮すること。

ハイウォーター・マークを無効にするためには、ボリュームを初期化するとき /NOHIGHWATER 修飾子を指定する。あるいは、まず、DCL の SET VOLUME コマンドを次の形式で実行する。

SET VOLUME/NOHIGHWATER_MARKING # デバイス指定 [:]

- OpenVMS レコード管理サービス (RMS) のファイル拡張パラメータの設定。

シーケンシャル・ファイルはマルチブロック数 (省略時の値は 32) の 2 倍単位で拡張されるので、省略時の設定では、ディスクのクラスタ・サイズに最も近くなるまで 64 ブロックずつ拡張される (ファイルの拡張についての詳細は、『Guide to OpenVMS File Applications』を参照)。ファイルが作成または拡張される際には、入出力動作が増えるため、応答速度が低下することがある。この問題は、ファイル拡張パラメータの値を大きくするか、システム・パラメータ RMS_EXTEND_SIZE を設定すれば解決する。システム・パラメータの変更方法については、1.5 項「AUTOGEN によるシステム・パラメータの変更」を参照。システム・パラメータの一覧については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照。

ファイル拡張の詳細については、『Guide to Creating OpenVMS Modular Procedures』を参照。ただし、現在は多数のネットワーク記憶デバイスが利用可能なため、他のガイドライン (逆効果のガイドラインまでも) が存在する可能性がある。

- 頻繁に使用するイメージのインストール。

複数のプロセスからルーチン単位で同時にアクセスされるイメージは、INSTALL ユーティリティで修飾子として /OPEN, /SHARED, および /HEADER_RESIDENT を指定してインストールする。そうすれば、すべてのプロセスがそのイメージの同じ物理コピーを使用し、そのイメージが最も効率的な方法で起動されるようになる。

通常、修飾子 /OPEN, /HEADER_RESIDENT, および /SHARED を付けてインストールすると、イメージにはおよそ 2 つの物理ページが追加される。INSTALL ユーティリティの

LIST/FULL コマンドを使用すると、/SHARED 修飾子を使ってインストールされたイメージへの最大同時アクセス数が示される。この値は、イメージのインストールがメモリの効果的な使用につながっているかどうかの判断に役立つ。

イメージのインストールについては、3.10.11 項「INSTALL によるイメージのインストール」と、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の INSTALL の節を参照。

- 共用可能イメージと実行可能イメージのインストール (Alpha システムおよび I64 システムの場合)。インストール・ユーティリティで /RESIDENT 修飾子を指定する。詳細は 3.10.6 項「共用アドレス・データを付けたイメージのインストール」を参照。
これは CPU とメモリのどちらをとるかという問題になる。/RESIDENT 修飾子を付けてイメージをインストールするということは、コードがページングされないということを意味する。共用イメージの量によって、メモリが増えることも減ることもあり得る。
- システム・ディスクの入出力の軽減。
頻繁にアクセスされるファイル(以下に例を示す)をシステム・ディスクから移動し、論理名を使用して記憶位置あるいは必要であれば他のポインタを指定する。
 - SYSUAF.DAT (SYSUAF は論理名)
 - RIGHTS.LIST.DAT (RIGHTS.LIST は論理名)
 - VMMAIL_PROFILE.DAT (VMMAIL_PROFILE は論理名)
 - NETPROXY.DAT (NETPROXY は論理名)
 - NET\$PROXY.DAT (NET\$PROXY は論理名)
 - キュー・データベース (詳細は『OpenVMS システム管理者マニュアル (上巻)』を参照)
 - ERRFMT ログ・ファイル (SYS\$ERRORLOG は論理名)
 - MONITOR ログ・ファイル (SYS\$MONITOR は論理名)
 - 会計情報ログ・ファイル (ACCOUNTING は論理名)
 - SECURITY_AUDIT.AUDIT\$JOURNAL (SET AUDIT/JOURNAL=SECURITY/DESTINATION=**ファイル指定**)
 - 省略時の DECnet for OpenVMS アカウント (OpenVMS ディストリビューション・キットの SYSUAF ファイルに含まれているレコード)以上のシステム・ファイルの論理名を再定義する場合は、サイト別コマンド・プロシージャ SYS\$MANAGER:SYLOGICALS.COM を変更する。SYLOGICALS.COM で論理名を定義する方法については、『OpenVMS システム管理者マニュアル (上巻)』を参照。
ページングやスワッピングの動作をシステム・ディスクからアクセス頻度の低い別のディスクに移すのも、システム・ディスクの入出力を軽減する方法の 1 つである。その場合は、移動先のディスクに大きな 2 次ページ・ファイルおよび 2 次スワップ・ファイルを作成する。しかし、システム障害を診断するためのクラッシュ・ダンプを保存したい場合は、システム・ディスク上のシステム固有のディレクトリ SYS\$SPECIFIC:[SYSEXE] にダンプ・ファイルを格納する。SYS\$SPECIFIC:[SYSEXE] にダンプ・ファイルが存在しない場合、クラッシュ・ダンプを保存するためには 1 次ページ・ファイルが必要になる。ページ・ファイルおよびスワップ・ファイルの移動方法については、2.16 項「ページ・ファイル、スワップ・ファイル、ダンプ・ファイルの作成と変更」を参照。
- Alpha システムおよび I64 システムでは、メモリ常駐セクションで使用するために、大量のメモリを確保しておく必要がある。予約メモリ・レジストリでは、そのインタフェースから、SYSMAN ユーティリティで、OpenVMS Alpha システムまたは I64 システムが、メモリ常駐セクションで使用したり、特権を持つ他のアプリケーションで使用するための、大量のメモリを確保するように設定することができる。さらに、予約メモリ・レジストリを使用することにより、割り当て済みの予約メモリを考慮しながら、AUTOGEN ユーティリティによって OpenVMS システムを適切にチューニングすることができる。詳細は、3.11 項「予約メモリ・レジストリ」を参照。

3.9 システム・ライブラリの展開

OpenVMS オペレーティング・システムに付属する大規模なシステム・ライブラリの一部は、ディスク使用量を抑えるために縮小(圧縮)されています。これらのライブラリを展開(つまり復元)すると、システムはこれらのライブラリにより短時間でアクセスできるようになりますが、使用するディスク領域は増加します。システム・ライブラリの一部またはすべてを展開するために十分なディスク領域があるかどうかを判断する方法については、3.9.1 項「ライブラリを展開するために使用可能なディスク領域の決定」を参照してください。

既存のライブラリに対する削除アクセスがあり、ライブラリ・ディレクトリに対する書き込みアクセスがある場合には、以下のいずれかの方法でライブラリを展開したり、縮小(圧縮)することができます。

- LIBDECOMP.COM コマンド・プロシージャを実行する(3.9.2 項「ライブラリ復元ユーティリティ (LIBDECOMP.COM) の使用」を参照)。データ縮小(圧縮)形式で付属するライブラリのうち、選択したライブラリから1つ、複数、またはすべてを、展開または縮小することができるようになる。
- DCL コマンド LIBRARY に /DATA 修飾子(3.9.3 項「LIBRARY コマンドでの /DATA 修飾子の使用」を参照)を付けて入力し、一度に1つずつ、データ縮小形式で付属していないライブラリを含めて、ライブラリを展開または縮小する。



注意:

本書では、“復元”および“圧縮”は、LIBDECOMP.COM プロシージャまたは LIBRARY コマンドに /DATA=EXPAND 修飾子または /DATA=REDUCE 修飾子を付けて使用した結果実行される処理を意味します。これらの処理を、/COMPRESS 修飾子を指定した結果実行される、全く異なる関係のない機能と混同しないように注意してください。LIBRARY コマンドとその修飾子についての詳細は、『OpenVMS Command Definition, Librarian, and Message Utilities Manual』またはオンライン・ヘルプを参照してください。

3.9.1 ライブラリを展開するために使用可能なディスク領域の決定

ライブラリを展開する前には、展開されたファイルに対応するための十分な未使用ディスク領域がシステムにあることを確認します。未使用ディスク領域のサイズを確認するには、以下のコマンドを入力します。

```
§ SHOW DEVICE SYS$SYSDEVICE
```

データ縮小形式で付属するライブラリすべてを拡張するために必要なディスク領域のサイズは、オペレーティング・システムおよび設定によって異なります。Alpha システムおよび I64 システムの場合には、約 60,000 ブロックの未使用領域が必要です。VAX システムの場合には、必要な未使用領域のサイズはこれよりも小さくなります。特定のディスクの必要条件については、対応するオペレーティング・システムの『Upgrade and Installation Manual』を参照してください。

必要な未使用ディスク領域がない場合、またはシステム・ライブラリをすべて展開しない場合、一部を選んで展開することができます。たとえば、システム・ヘルプ・ライブラリ (HELPLIB.HLB) と、システムで頻繁に使用する他のライブラリだけを展開して、他のライブラリをデータ縮小形式のままにしておくことができます。

3.9.2 ライブラリ復元ユーティリティ (LIBDECOMP.COM) の使用

ライブラリ復元ユーティリティはコマンド・プロシージャ LIBDECOMP.COM であり、SYS\$UPDATE ディレクトリにあります。LIBDECOMP.COM は LIBRARY コマンドを使用して、データ縮小形式で付属するシステム・ライブラリの一部またはすべてを、展開(復元)または縮小(圧縮)します(このようなライブラリのリストについては、3.9.2.1 項「LIBDECOMP.COM が動作するライブラリ」を参照してください)。

LIBDECOMP.COM コマンド・プロシージャを使用すると、複数の LIBRARY コマンドではなく、1つのコマンドを入力して、一度にすべてまたは多くのライブラリを展開または縮小すること

ができます。LIBDECOMP.COM は、会話型でも (3.9.2.2 項「LIBDECOMP.COM の会話型での使用」を参照) バッチ・モードでも (3.9.2.3 項「バッチ・モードでの LIBDECOMP.COM の使用」を参照) でも実行することができます。

3.9.2.1 LIBDECOMP.COM が動作するライブラリ

ライブラリ復元ユーティリティは、一部の大規模なライブラリ、つまりデータ縮小形式で付属するライブラリで動作します。ライブラリによってはプラットフォーム固有で、VAX システムだけに付属するものや、Alpha システムや I64 システムに付属するものがあります。ライブラリの多くはオプションのコンポーネントであり、OpenVMS のインストール時にすべてのオプションが選択された場合以外は、システムによっては存在しないこともあります。表 3-1 「LIBDECOMP.COM が認識できるライブラリ」は、VAX システムと、Alpha システムや I64 システムで LIBDECOMP.COM が認識できるすべてのライブラリをリストしています。

表 3-1 LIBDECOMP.COM が認識できるライブラリ

ライブラリ名	プラットフォーム	説明
[SYSHLP] ディレクトリのヘルプ・ライブラリ・ファイル (.HLB)		
ACLEDT.HLB	すべて	アクセス制御リスト・エディタ・ヘルプ
BKM\$HELP.HLB	すべて	バックアップ・マネージャ・ヘルプ
DBG\$HELP.HLB	すべて	OpenVMS デバッガ・ヘルプ
DBG\$UIHELP.HLB	すべて	OpenVMS デバッガ・ヘルプ
EDTHELP.HLB	すべて	EDT エディタ・ヘルプ
EVE\$HELP.HLB	すべて	EVE エディタ・ヘルプ
EVE\$KEYHELP.HLB	すべて	EVE キーボード・ヘルプ
EXCHNGHLP.HLB	すべて	Exchange ユーティリティ・ヘルプ
HELPLIB.HLB	すべて	DCL ヘルプ
LANCP\$HELP.HLB	すべて	LAN 制御プログラム・ヘルプ
LATCP\$HELP.HLB	すべて	LAT 制御プログラム・ヘルプ
MAILHELP.HLB	すべて	メール・ユーティリティ・ヘルプ
NCPHELP.HLB	すべて	ネットワーク制御プログラム・ヘルプ
SDA.HLB	すべて	System Dump Analyzer ヘルプ
SHWCLHELP.HLB	すべて	Show Cluster ユーティリティ・ヘルプ
SYSGEN.HLB	すべて	System Generation ユーティリティ・ヘルプ
SYSMANHELP.HLB	すべて	System Management ユーティリティ・ヘルプ
TPUHELP.HLB	すべて	Text Processing ユーティリティ・ヘルプ
UAFHELP.HLB	すべて	Authorize ユーティリティ・ヘルプ
[SYSLIB] ディレクトリのマクロ・ライブラリ・ファイル (.MLB)		
LANIDF.MLB	Alpha または I64	LAN 内部ドライバ・マクロ
LIB.MLB	すべて	オペレーティング・システム・マクロ
STARLET.MLB	すべて	オペレーティング・システム・マクロ
SYBLDMLB.MLB	VAX のみ	システム構築マクロ
[SYSLIB] ディレクトリのオブジェクト・ライブラリ・ファイル (.OLB)		

表 3-1 LIBDECOMP.COM が認識できるライブラリ (続き)

ライブラリ名	プラットフォーム	説明
DECCRTL.OLB	VAX のみ	HP C 実行時ライブラリ
STARLET.OLB	すべて	システム・オブジェクト・ライブラリおよび実行時ライブラリ
SYSBLDLIB.OLB	VAX のみ	システム構築ファイル
VAXCRTL.OLB	すべて	HP C RTL ルーチン名エントリ・ポイント。 VAX G 浮動小数点倍精度エントリ・ポイント および浮動小数点エントリ・ポイント。
VAXCRTLD.OLB	Alpha または I64	VAX D 浮動小数点倍精度エントリ・ポイント および浮動小数点エントリ・ポイントの 一部サポート
VAXCRTLDX.OLB	Alpha または I64	VAX D 浮動小数点のサポート。 /L_DOUBLE_SIZE=128 コンパイラ修飾子の サポート
VAXCRTLT.OLB	Alpha または I64	IEEE T 浮動小数点倍精度エントリ・ポイント、 浮動小数点エントリ・ポイント
VAXCRTLTX.OLB	Alpha または I64	IEEE T 浮動小数点のサポート。 /L_DOUBLE_SIZE=128 コンパイラ修飾子の サポート
VAXCRTLX.OLB	Alpha または I64	G 浮動小数点のサポート。 /L_DOUBLE_SIZE=128 コンパイラ修飾子の サポート
VMS\$VOLATILE_PRIVATE_INTERFACES.OLB	Alpha または I64	OpenVMS バグチェック処理コード
[SYSLIB] ディレクトリのテキスト・ライブラリ・ファイル (.TLB)		
ERFLIB.TLB	すべて	ANALYZE/ERROR デバイス記述
LIB_ADA_SUBSET.TLB	すべて	Ada プログラム・ツールキットのオペレー ティング・システムの定義
NTA.TLB	すべて	NTA 機能に対して構築するファイル
STARLET_RECENT_ADA_SUBSET.TLB	すべて	Ada プログラム・ツールキットのオペレー ティング・システムの定義
STARLETSD.TLB	すべて	レイヤード・プロダクトのインストール中 に使用する STARLET の定義
SYS\$LIB_C.TLB	Alpha または I64	C 言語のヘッダ・ファイル。LIB から派生。
SYS\$STARLET_C.TLB	すべて	HP C のための公用ヘッダ・ファイル

LIBDECOMP.COM コマンド・プロシージャのリスト関数を使用すれば、ライブラリ復元ユーティリティが認識できるすべてのライブラリのリストを、そのサイズおよびシステムでの状態を含めて出力することができます。詳細および例は、3.9.2.2.1 項「ライブラリのリスト」を参照してください。

3.9.2.2 LIBDECOMP.COM の会話型での使用

ライブラリ復元ユーティリティを実行するための基本的なコマンドは、次のとおりです。

```
@ SYS$UPDATE:LIBDECOMP [parameters]
```

ライブラリ復元ユーティリティでは、最大で 8 つのオプションのパラメータを使用できます。最初のパラメータは、このユーティリティの 3 つの機能のうち、どれを実行するかを制御しま

す。他のパラメータは、このユーティリティがどのライブラリを処理するかを制御します。これら 3 つの機能については、この後のセクションで説明します。

- リスト機能 (3.9.2.2.1 項「ライブラリのリスト」を参照)
ライブラリ復元ユーティリティが認識できるすべての VAX および Alpha または 164 ライブラリをリストし、現在使用しているシステムにあるライブラリのサイズおよび状態 (縮小されているか展開されているか) を表示します。
- 展開機能 (3.9.2.2.2 項「ライブラリの展開 (復元)」を参照)
データ縮小形式のライブラリを展開します。これは、省略時の機能です。
- 縮小機能 (3.9.2.2.3 項「ライブラリの縮小 (圧縮)」を参照)
展開されたライブラリを、データ縮小形式に圧縮します。

簡単なオンライン・ヘルプを表示するには、次のコマンドを入力します。

```
$ @SYS$UPDATE:LIBDECOMP HELP
```

3.9.2.2.1 ライブラリのリスト

ライブラリ復元ユーティリティが認識できるすべての VAX および Alpha または 164 ライブラリを、それらのサイズおよびシステムでの状態もともにリストするには、次のようにコマンドの中で LIST パラメータを指定します。

```
$ @SYS$UPDATE:LIBDECOMP LIST
```

コマンドを実行した結果表示されるリストには、VAX システムや Alpha システムまたは 164 システムに付属していないため、または関連付けられている機能がシステムにインストールされていないために、現在使用しているシステムにはないライブラリがどれであるかが示されません。現在使用しているシステムにあるライブラリについては、そのサイズおよび現在の状態 (縮小されているか展開されているか) がリストに表示されます。次の例は、Alpha システムまたは 164 システムでの出力例を示しています (注: ファイル・サイズは変わる可能性があります。最も正確な情報については、現在使用しているシステムでの LIST の出力を参照してください)。

```
$ @SYS$UPDATE:LIBDECOMP LIST
```

```
OpenVMS Library Decompression Utility
List of all libraries known to LIBDECOMP
"Library not present" indicates not installed on this system
```

Libraries in SYS\$SYSROOT:

	Library	Size	
1)	[SYSHLP] ACLEDT.HLB	70	Reduced format
2)	[SYSHLP] BKM\$HELP.HLB	156	Reduced format
3)	[SYSHLP] DBG\$HELP.HLB	1234	Reduced format
4)	[SYSHLP] DBG\$UIHELP.HLB	269	Reduced format
5)	[SYSHLP] EDTHELP.HLB	154	Reduced format
6)	[SYSHLP] EVE\$HELP.HLB	676	Reduced format
7)	[SYSHLP] EVE\$KEYHELP.HLB	99	Reduced format
8)	[SYSHLP] EXCHNGHLP.HLB	83	Reduced format
9)	[SYSHLP] HELPLIB.HLB	9179	Reduced format
10)	[SYSHLP] LANCP\$HELP.HLB	119	Reduced format

11)	[SYSHLP] LATCP\$HELP.HLB	157	Reduced format
12)	[SYSHLP] MAILHELP.HLB	211	Reduced format
13)	[SYSHLP] NCPHELP.HLB	261	Reduced format
14)	[SYSHLP] SDA.HLB	308	Reduced format
15)	[SYSHLP] SHWCLHELP.HLB	103	Reduced format
16)	[SYSHLP] SYSGEN.HLB	337	Reduced format
17)	[SYSHLP] SYSMANHELP.HLB	492	Reduced format
18)	[SYSHLP] TPUHELP.HLB	575	Reduced format
19)	[SYSHLP] UAFHELP.HLB	241	Reduced format
20)	[SYSLIB] LANIDEF.MLB	181	Reduced format
21)	[SYSLIB] LIB.MLB	2715	Reduced format
22)	[SYSLIB] STARLET.MLB	2335	Reduced format
23)	[SYSLIB] SYSBLDMLB.MLB		Library not present
24)	[SYSLIB] DECCRTL.OLB		Library not present
25)	[SYSLIB] STARLET.OLB	27461	Reduced format
26)	[SYSLIB] SYSBLDLIB.OLB		Library not present
27)	[SYSLIB] VAXCRTL.OLB	1163	Reduced format
28)	[SYSLIB] VAXCRTLD.OLB	1587	Reduced format
29)	[SYSLIB] VAXCRTLDX.OLB	1506	Reduced format
30)	[SYSLIB] VAXCRTLT.OLB	1434	Reduced format
31)	[SYSLIB] VAXCRTLTX.OLB	1449	Reduced format
32)	[SYSLIB] VAXCRTLX.OLB	1285	Reduced format
33)	[SYSLIB] ERFLIB.TLB	64	Reduced format
34)	[SYSLIB] LIB_ADA_SUBSET.TLB	1839	Reduced format
35)	[SYSLIB] NTA.TLB	34	Reduced format
36)	[SYSLIB] STARLETS.D.TLB	3940	Reduced format
37)	[SYSLIB] SYS\$LIB_C.TLB	9442	Reduced format
38)	[SYSLIB] SYS\$STARLET_C.TLB	5864	Reduced format
39)	[SYSLIB] VMS\$VOLATILE_PRIVATE_INTERFACES.OLB	445	Reduced format
40)	[SYSLIB] STARLET_RECENT_ADA_SUBSET.TLB	1100	Reduced format
Total Libraries: 37		78568	

3.9.2.2.2 ライブラリの展開 (復元)

LIBDECOMP.COM の省略時の処理は、展開機能です。EXPAND をコマンド行の最初のパラメータとして指定することもできますが、特に必要はありません。最初のパラメータが LIST または REDUCE である場合を除いて、省略時には展開機能が使用されます。

展開機能が使用される場合には、残りのパラメータは、どのライブラリが展開されるかを指定します。ALL を指定して、ライブラリ復元ユーティリティが認識できるすべてのライブラリを展開することもできますが、ライブラリ名を 8 つまで (EXPAND を指定した場合には 7 つまで) 指定することもできます。ワイルドカード文字は、許可されていません。指定されるライブラリは、このユーティリティが認識できるものでなければなりません (他のどのライブラリも、展開するには、3.9.3 項「LIBRARY コマンドでの /DATA 修飾子の使用」で説明するように、LIBRARY コマンドを使用しなければなりません)。ALL またはライブラリのリストを指定しない場合、LIBDECOMP.COM は、展開するライブラリを選択するよう指示するプロンプトを表示します。このプロンプトに対しては、いくつでもライブラリを指定することができます。



注意:

LIBDECOMP.COM が認識できるすべてのライブラリを展開または縮小するには、通常およそ 5 ~ 10 分かかります。ただし、現在使用しているシステムのハードウェアおよびソフトウェア構成、およびその他の実行状態によっては、最大で 30 分以上になる場合もあるなど、さらに長くかかることがあります。

展開機能を使用すると、次の OpenVMS Alpha の例で表示される内容に似たものが表示されず。ヘッダ行が表示された後、LIBDECOMP.COM がそれぞれのライブラリをチェックする間、一時停止します。

```
OpenVMS Library Decompression Utility
Candidate Libraries to be expanded
(Libraries not present and libraries already expanded are not listed)

  1  ACLEDT.HLB          13  NCPHELP.HLB          25  VAXCRTLD.OLB
  2  BKM$HELP.HLB       14  SDA.HLB              26  VAXCRTLDX.OLB
  3  DBG$HELP.HLB       15  SHWCLHELP.HLB        27  VAXCRTLT.OLB
  4  DBG$UIHELP.HLB     16  SYSGEN.HLB           28  VAXCRTLTX.OLB
  5  EDTHelp.HLB        17  SYSMANHELP.HLB       29  VAXCRTLX.OLB
  6  EVE$HELP.HLB       18  TPUHELP.HLB          30  ERFLIB.TLB
  7  EVE$KEYHELP.HLB    19  UAFHELP.HLB          31  LIB_ADA_SUBSET.TLB
  8  EXCHNGHLP.HLB      20  LANIDF.MLB           32  NTA.TLB
  9  HELPLIB.HLB        21  LIB.MLB              33  STARLETSD.TLB
 10  LANCP$HELP.HLB     22  STARLET.MLB          34  SYS$LIB_C.TLB
 11  LATCP$HELP.HLB     23  STARLET.OLB          35  SYS$STARLET_C.TLB
 12  MAILHELP.HLB      24  VAXCRTL.OLB

      36  VMS$VOLATILE_PRIVATE_INTERFACES.OLB
      37  STARLET_RECENT_ADA_SUBSET.TLB

      A  ALL libraries to be expanded
      H  Display HELP information for LIBDECOMP
      E  EXIT this procedure
```

ALL を指定した場合、次のメッセージが表示され、ライブラリ復元ユーティリティは、リストされているすべてのライブラリを展開します。

```
"ALL" specified; all libraries will be processed
```

ALL を指定しない場合、次のプロンプトが表示されます。

```
* Enter a letter or the number(s) of libraries to be expanded
  (Separate multiple numbers with a comma)
```

A, H, E, または 1 つまたは複数の数字を入力して、展開するライブラリを指定します。指定できる数字の制限はありません。

展開する特定のライブラリを識別するパラメータを入力した場合、LIBDECOMP.COM が実行されても、例で示されているようなりストは表示されません。それぞれのライブラリは、処理されるたびにリストされます。

例

- 次のいずれかのコマンドを実行すると、すべてのライブラリが展開されます。

```
$ @SYS$UPDATE:LIBDECOMP ALL
$ @SYS$UPDATE:LIBDECOMP EXPAND ALL
```

- メニューにある一部またはすべてのライブラリを展開するかどうかを選択するには、まず、次のいずれかのコマンドを入力します。

```
$ @SYS$UPDATE:LIBDECOMP
$ @SYS$UPDATE:LIBDECOMP EXPAND
```

次に、以前の例で示したとおり、すべてのライブラリを指定するか、または展開する個々のライブラリの番号をいくつでも指定するよう、プロンプトが表示されます。

- このメニューを省略し、選択したライブラリだけを展開するには、次のいずれかのコマンドのようなコマンドを入力して、プロンプトに回答する代わりに、コマンド行にライブラリ名を指定します。

```
$ @SYS$UPDATE:LIBDECOMP HELPLIB.HLB STARLET.MLB LIB.MLB
$ @SYS$UPDATE:LIBDECOMP EXPAND HELPLIB.HLB STARLET.MLB LIB
```

どちらのコマンドを実行しても、結果は同じです。HELPLIB.HLB ライブラリ、STARLET.MLB ライブラリ、および LIB.MLB ライブラリが展開されます。

3.9.2.2.3 ライブラリの縮小 (圧縮)

コマンド行で最初のパラメータとして REDUCE を指定して LIBDECOMP.COM を実行すると、展開されていたライブラリが縮小されます。

REDUCE の後に ALL を指定して、ライブラリ復元ユーティリティが認識できるすべてのライブラリを縮小することも、最大で 7 つまでのライブラリを指定することもできます。ワイルドカード文字は、許可されていません。

ALL または少なくとも 1 つのライブラリの名前を指定しないと、LIBDECOMP.COM は、縮小するライブラリの名前を入力するようプロンプトを表示します。このプロンプトに対してリストできるライブラリの数に制限はありません。

縮小機能を使用すると、展開機能を使用した場合に似た内容が表示されますが、縮小できるライブラリだけが表示される点が異なります。

例

- 次のコマンドを実行すると、展開されていたすべてのライブラリが縮小されます。

```
$ @SYS$UPDATE:LIBDECOMP REDUCE ALL
```

- メニューにある一部またはすべてのライブラリを縮小するかどうかを選択するには、まず、次のいずれかのコマンドを入力します。

```
$ @SYS$UPDATE:LIBDECOMP REDUCE
```

次に、展開機能のセクションで示されたとおり、すべてのライブラリを指定するか、または展開する個々のライブラリの番号をいくつでも指定するよう、プロンプトが表示されます。

3.9.2.3 バッチ・モードでの LIBDECOMP.COM の使用

次のように、DCL コマンド SUBMIT に /PARAMETERS 修飾子を使用することによって、ライブラリ復元ユーティリティをバッチ・キューに登録することができます。

```
SUBMIT SYS$UPDATE:LIBDECOMP /PARAMETERS=(p1[,p2,...])
```

バッチ・プロシージャを実行すると会話型プロシージャと同じ結果が生成されますが、バッチ・ジョブの場合、HELP, LIST, ALL または少なくとも 1 つのライブラリ名を指定する必要があります。バッチ・ジョブは、入力するようプロンプトを表示することができないためです。

パラメータは最大で 8 つまで指定することができます。複数のパラメータを指定した場合、それらのパラメータは丸括弧で囲み、それぞれをコンマで区切る必要があります。

例

- 次のコマンドを実行すると、LIBDECOMP.COM が認識できるすべてのライブラリのリストが出力されます。

```
$ SUBMIT SYS$UPDATE:LIBDECOMP /PARAMETERS=LIST
```

- 省略時には、次のコマンドを実行すると、該当するすべてのコマンドが展開されます。

```
$ SUBMIT SYS$UPDATE:LIBDECOMP /PARAMETERS=ALL
```

- 次のコマンドを実行すると、HELPLIB.HLB, STARLET.MLB, および LIB.MLB ライブラリが展開されます。

```
$ SUBMIT SYS$UPDATE:LIBDECOMP -  
_ $ /PARAMETERS=(EXPAND,HELPLIB.HLB,STARLET.MLB,LIB.MLB)
```

3.9.3 LIBRARY コマンドでの /DATA 修飾子の使用

ライブラリ復元ユーティリティを使用せずに個々のライブラリを展開または縮小する方法としては、DCL コマンド LIBRARY に /DATA 修飾子を使用する方法があります。この方法では、1 つの LIBRARY コマンドで指定できるライブラリは 1 つだけです。

LIBDECOMP.COM は、データ縮小形式で付属する約 40 個のライブラリに対してのみ実行できますが、LIBRARY コマンドは、ほとんどどのライブラリ・ファイルでも、展開または縮小することができます。ただし、ライブラリ・タイプ(ファイル拡張子)は、OpenVMS Librarian ユーティリティが認識できるものでなければなりません。以下のシステム・ライブラリは、縮小しないでください。

- [SYSLIB]EPC\$FACILITY.TLB
このファイルは、空のまま付属しています。
- [SYSLIB]IMAGELIB.OLB
このライブラリに対する縮小処理は無効です。
- [SYSLIB]NCS\$LIBRARY.NLB
OpenVMS Librarian ユーティリティは、ライブラリ・タイプ .NLB を認識できません。

指定されたライブラリを展開するには、次のコマンド形式を使用します。

```
LIBRARY library-name.ext /DATA=EXPAND
```

指定されたライブラリを縮小するには、次のコマンド形式を使用します。

```
LIBRARY library-name.ext /DATA=REDUCE
```

ライブラリの拡張子 (.HLB, .MLB, .OLB, または .TLB) は、常に指定しなければなりません。

指定されたライブラリが現在の省略時のデバイスおよびディレクトリにない場合には、ライブラリ指定の中でデバイスおよびディレクトリも指定しなければなりません。ほとんどのシステム・ライブラリは、次の例のように、SYS\$HELP ([SYSHLP]) または SYS\$LIBRARY ([SYSLIB]) にあります。

\$ LIBRARY [SYSHLP]HELPLIB.HLB /DATA=EXPAND

LIBRARY コマンドのその他の修飾子についての詳細は、LIBRARY のオンライン・ヘルプまたは『OpenVMS Command Definition, Librarian, and Message Utilities Manual』を参照してください。

3.10 INSTALL による既知イメージのインストール

Install ユーティリティ (INSTALL) は、イメージに関する情報をメモリに格納します。INSTALL は次の目的で使用します。

目的	参照箇所
同時に使用されるイメージが消費するメモリを節約する	3.10.7 項 「物理メモリを節約するイメージのインストール方法」
システム性能を向上させる	3.10.5 項 「イメージ起動性能を向上するためのイメージのインストール方法」
Alpha システムおよび I64 システムにおいて、共用アドレス・データのあるイメージを使用することで性能を向上させる ¹	3.10.6 項 「共用アドレス・データを付けたイメージのインストール」
拡張特権が必要な実行可能イメージを一般的に利用可能にする	3.10.8.1 項 「特権付き実行可能イメージ」
非特権イメージが、共用イメージの特権機能呼び出せるようにする	3.10.8.2 項 「特権付き共用可能イメージ」
共用イメージが特権実行可能イメージにより起動できるように保護マークを付ける	3.10.9 項 「特権コンテキストでのイメージの起動方法」

¹ Alpha または I64 のみ

サイト別スタートアップ・コマンド・プロシージャ STARTUP.COM により、システム・ブート時に INSTALL を使用していくつかのシステム・プログラムがインストールされます。それ以外のプログラムを必要に応じてインストールする場合は、INSTALL を使用します。

このようにメモリにインストールして使用するイメージ(インストール済みイメージ)は、システムのリブートのたびにインストールしなおす必要があります。そのため、サイト別スタートアップ・コマンド・プロシージャ SYSTARTUP_VMS.COM に必要な INSTALL コマンドを追加します。詳細は『OpenVMS システム管理者マニュアル (上巻)』を参照してください。

Install ユーティリティ (INSTALL) は、/NOTRACEBACK 修飾子でリンクされたイメージだけをインストールします。

INSTALL コマンドの機能は SYSGEN ユーティリティの INSTALL コマンドと異なる点に注意してください。

次に、インストール済みイメージの概念と Install ユーティリティの使用法を説明します。

3.10.1 イメージおよび既知イメージについて

イメージとは、実行可能プログラムを形成するために Linker ユーティリティによって結合されたプロシージャとデータの集まりです。実行可能イメージはプロセス内で、あるいはコマンド・ライン・インタプリタ (CLI) または \$CREPRC システム・サービスによって、実行できません。通常、実行可能プログラムのファイル・タイプは .EXE です。

イメージには次の 3 つのタイプがあります。

イメージ・タイプ	説明
実行可能	リンカで /EXECUTABLE 修飾子を指定して (あるいは /SHAREABLE 修飾子を指定しないで) リンクされたイメージ。詳細は『OpenVMS Linker Utility Manual』を参照。
共用可能	linker ユーティリティで /SHAREABLE 修飾子を指定してリンクされたイメージ。共用可能イメージは、別のファイルのリンクの入力ファイルとして、暗黙的あるいは明示的に指定できることから、リンク可能イメージと呼ばれることもある。共用可能イメージは、それにリンクする実行可能イメージにはコピーされない。したがって、リンク実行可能イメージの数にかかわらず、ディスク上には共用可能イメージが 1 つだけ存在すればよい。詳細は『OpenVMS Linker Utility Manual』を参照。
システム	オペレーティング・システムの制御のもとで実行しないイメージ。スタンドアロン動作だけを目的とする。システム・イメージの内容と形式は、共用可能イメージおよび実行可能イメージと異なる。詳細は『OpenVMS Linker Utility Manual』を参照。

INSTALL によってイメージをインストールすると、そのイメージには属性が割り当てられ、システムにとって「既知」のものになります。このため、インストール済みイメージは**既知イメージ**とも呼ばれます。

イメージ・アクティベータは、既知イメージを優先するために、検索リストを 2 つのパスで処理します。検索リストの 1 回目の検索で、イメージ・アクティベータは既知ファイルとしてイメージを探します。必要な場合、検索リストの 2 回目の検索で、イメージ・アクティベータはディスク上でイメージを探します。

3.10.2 既知ファイルのエントリについて

システムは、既知イメージを**既知ファイル・エントリ**と呼ばれる内部データ構造に定義します。各エントリは、インストールされたイメージの名前と、インストール時に割り当てられた属性を示します (インストール済みイメージの属性についての詳細は、3.10.3 項「既知イメージに割り当てることができる属性」を参照してください)。

既知ファイル・エントリは、システムが稼働している間だけ存在します。システムがシャットダウンしたり、何らかの理由で異常終了した場合には、システムのリポート後にすべての既知イメージを再インストールする必要があります。

3.10.3 既知イメージに割り当てることができる属性

INSTALL コマンドに修飾子を指定すれば、既知イメージに属性を割り当てることができます。表 3-2「既知イメージの属性」に、既知イメージに割り当てることができる属性と使用される修飾子を示します。

表 3-2 既知イメージの属性

属性	説明	修飾子
ヘッダ常駐	イメージ・ファイルのヘッダ(ネイティブ・イメージのみ)がメモリに常駐したままとなるため、ファイルに 1 回アクセスするために行われるディスク入出力が 1 回少なくなる。ヘッダが 1 ブロックだけで構成される場合、1 つのファイルあたり 512 バイトのページング動的メモリが使用される。ヘッダが複数のブロックで構成される場合、1 つのファイルあたりのメモリ消費量はヘッダのブロック数によって異なる。ヘッダ・オープンとしてインストールされたイメージは、暗黙的に永久オープンとしてインストールされる。	/[NO]HEADER_RESIDENT
永久オープン	イメージ・ファイルはオープンしたままとなる。そのため、そのイメージへのアクセスには、ファイル・システムを呼び出す必要がない。	/OPEN

表 3-2 既知イメージの属性 (続き)

属性	説明	修飾子
特権	イメージにこの属性を割り当てると、そのイメージを実行するすべてのプロセスに一時的に強い特権が割り当てられ、イメージの実行中には利用者登録ファイル (UAF) で指定された特権の制限を超える処理を行うことができる。したがって、通常の特権を持つユーザでも、通常より上の特権が必要となるプログラムを実行できる。この属性(およびそれを作成するための /PRIVILEGED 修飾子)は、実行可能イメージだけに適用できる。	/PRIVILEGED[=(privilege,...)]
保護	イメージが起動されると、そのイメージのアドレス領域は、ユーザ・モードのコードによる変更から保護される。このことは、カーネルまたはエグゼクティブ・モードで実行する共用可能コードには重要である。	/PROTECTED
常駐 ¹	Alpha システムおよび I64 システムの場合、イメージのコードまたは読み込み専用のデータは、メモリのシステム領域に永久に常駐される。このため、変換バッファ (TB) のミス率を減らすための特殊なページ・マップを使用することで性能が向上する。常駐属性は、修飾子 /SECTION_BINDING=(CODE,DATA) にリンクされた共用可能イメージまたは実行可能イメージに適用される。	/RESIDENT
共用	イメージの読み込み専用セクションおよび参照時にコピーを行わない読み書き用セクションは、複数のユーザから同時にアクセスできる。したがって、そのようなセクションは物理メモリに 1 つだけ存在すればよい。一方、参照時にコピーを行うセクションは、それにアクセスする各プロセスごとに必要となる。共用イメージは、暗黙に「永久オープン」として宣言される。	/SHARED
書き込み可能	共用されている参照時にコピーを行わない書き込み可能セクションが物理メモリから削除される場合 (ページング機構により、あるいはそのセクションを参照するプロセスがないために)、そのセクションは同時にイメージ・ファイルに書き戻される。したがって、このセクションに行われた更新は保存され、初期値は失われる。このイメージは同時に「共用」としても宣言されなければならない。	/WRITABLE

¹ Alpha および I64 のみ

3.10.4 インストールするイメージの決定

次の条件を満たすイメージがインストールの対象となります。

- イメージの起動性能を向上させる
- Alpha システムおよび I64 システムの場合、共用イメージの実行性能を向上させる
- 物理メモリを節約する
- 実行中のイメージの特権を拡張する

ファイルをインストールするとページング動的メモリなどのシステム資源が必要になります。したがって、システム性能を向上させ、サイトの要件を満たすファイルをインストールします。INSTALL の LIST コマンドを利用すれば、イメージをインストールすることによって得られる利点を評価できます。たとえば、各イメージがアクセスされた回数を計算し、同時に行われているアクセスの数を示します。これらの値から、各イメージにオーバーヘッドの価値があるかどうかを判断できます。

3.10.5 イメージ起動性能を向上するためのイメージのインストール方法

頻繁に実行されるイメージをインストールすることにより、イメージの起動性能を向上できません。イメージの起動性能は、プログラムがインストールされている場合に向上しますが、これは、オペレーティング・システムがインストール済みファイルをファイル名ではなく、ファイル ID でオープンし、ディレクトリ操作が省略されるためです。

ヘッダ常駐としてイメージをインストールすると、システムがイメージ・ヘッダをメモリに読み込む場合の I/O 操作のオーバーヘッドを防止するため、起動性能がさらに強化されます。

ヘッダ常駐としてイメージをインストールするためには、インストール時に /HEADER_RESIDENT 修飾子を指定します。 /HEADER_RESIDENT を指定すると、イメージ・ファイルのヘッダが永久常駐し、ディスク I/O を減少させます。 /HEADER_RESIDENT 修飾子が指定されると、イメージは暗黙的に永久オープンになります。

イメージ・ヘッダは、ページング動的メモリに格納されます。イメージ・ヘッダのサイズはさまざまです。

頻繁にアクセスする重要なイメージは、オープンされたイメージとしてインストールすることができます。イメージを「永久オープン」としてインストールするためには、インストール時に /OPEN 修飾子を指定します。イメージ・ファイルはオープンしたままになるため、そのイメージへのアクセスには、ファイル・システムを呼び出す必要がなくなります。イメージ・ファイルを永久オープンにしておくには、1 ファイルごとに、約 512 バイトの非ページング動的メモリが必要になります。

3.10.6 共用アドレス・データを付けたイメージのインストール

OpenVMS Alpha システムおよび I64 システム上で共用アドレス・データを使用すると、次のときに性能が向上します。

- 実行時、共用アドレス・データは、プロセス間のメモリの共用度を高めるので、物理メモリを節約する。
- イメージの起動時、フィックスアップがインストレーション時に実行されるので、共用アドレス・データは CPU 時間と I/O 時間を減少させる。

詳細については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の INSTALL 項を参照してください。

関連項目

共用アドレス・データに関連する項目について、次で説明します。

- イメージ・セクション
1つのイメージは、いくつものイメージ・セクションから構成されている。1つのイメージ・セクションには、次のものが含まれる。
 - 命令 (コード)
 - 読み込み専用データ (定数)
 - 読み込み / 書き込み可能データ
- 共用既知イメージ
次のコマンドを使用すると、イメージを共用既知イメージにすることができる。

§ **INSTALL ADD** イメージ名 /SHARED

このコマンドを入力すると、Install ユーティリティは読み込み専用イメージ・セクション用にグローバル・セクションを作成し、イメージを実行するすべてのプロセスがこのセクションを共用できるようにする。

- アドレス・データ
イメージ・セクションの一種にはアドレス・データを含むものがある。アドレス・データ・セクションは実行時には読み込み専用になる。ただし、これらのアドレスはイメージ

の起動までは未知のものである。このため、イメージ・セクションは、イメージ起動の最後までは読み込み / 書き込みが可能である。共用イメージ用のアドレスは、マップされたイメージをさまざまなプロセスが独自に集める傾向があるので、一般に、プロセスによってさまざまである。

- 共用アドレス・データ

共用アドレス・データの機能は、P1 イメージ・セクションにある共用イメージに対して、独自の P1 空間アドレスを割り当てる (IMGREG_PAGES システム・パラメータが P1 空間のサイズを決定する)。独自の P1 空間アドレスを割り当てる (IMGREG_PAGES システム・パラメータが P1 空間のサイズを決定する)。割り当てられたアドレスによって、Install ユーティリティは、イメージをインストールする際に、アドレス・データ・セクションの内容を判断する。それぞれのアドレス・データ・イメージ・セクションに対して、共用アクセスができるように、グローバル・セクションが作成される。

実行可能な (メイン) イメージも、共用アドレス・データ・セクションを使用できる。ただし、これらのイメージは P1 アドレスには割り当てられない。その理由は、実行可能なイメージの基底アドレスは、イメージがリンクされるときに判断されるためである。

3.10.6.1 システムが提供するイメージ

OpenVMS ソフトウェア製品の一部である多くのイメージは、共用アドレス・データを伴う共用既知イメージとしてインストールされています。これによって、システム管理者が特に何かを行わなくても、性能は向上します。

3.10.6.2 アプリケーション・イメージ

システム管理者は、共用アドレス・データ付きで追加のイメージをインストールするように決定することもあります。これを考慮する際、アプリケーションの共用イメージに対する依存性を調査する必要があります。

3.10.7 物理メモリを節約するイメージのインストール方法

通常、複数のプロセスから同時に実行されるイメージをインストールすることにより、物理メモリを節約できます。イメージをインストールしていない場合、または共用属性を付けずにインストールした場合には、イメージを実行する各プロセスは、メモリ内のプライベート・セクションを必要とします。共用イメージは、コードのコピーが 1 つだけ常にメモリに存在し、多くのユーザが同時にそのコードにアクセスできるため、物理メモリを節約できます。イメージを共用イメージとしてインストールする場合には、/SHARED 修飾子を使用します。

共用属性を付けてイメージをインストールする場合は、パーマネント・システム・グローバル・セクションが作成されます。参照時にコピーを行わないグローバル・セクションを実行するときは、物理メモリに必要なコピーはセクションあたり 1 つだけです。このことは、そのセクションを持つイメージを実行するプロセスの数には影響されません。

共用属性付きでインストールできるイメージの数は、システム・パラメータ GBLPAGES および GBLSECTIONS によって制限されます。これらのシステム・パラメータについては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

3.10.8 イメージの特権を拡張するためのイメージのインストール方法

イメージを拡張特権環境で実行できるようにするためには、次の 2 つの方法があります。

- 既存の実行可能イメージを拡張特権付きでインストールし、特権を持たないプロセスがそのイメージの特権付きの機能を利用できるようにする。

INSTALL コマンドの CREATE または REPLACE に、/PRIVILEGED 修飾子を指定する。



重要:

拡張した特権を持つイメージをインストールすることにより、システムの機密保護が侵される場合があります。そのイメージにより、ユーザがその他の特権を使用できるような制御を得ることがないように確認してください。

- 特権付きの共用可能イメージ (ユーザが作成したシステム・サービスを実現するとき使用するもの) をインストールし、他の非特権イメージが、その特権を拡張しなくても特権付きコードの選択部分を実行できるようにする。
INSTALL コマンドの CREATE または REPLACE に、/PROTECTED および /SHARED 修飾子を指定する。

3.10.8.1 特権付き実行可能イメージ

特権を持たないプロセス (非特権プロセス) でも、特権イメージとしてインストールされた実行イメージの特権付き機能を利用することができます。そのためには、/PRIVILEGED 修飾子を指定して、該当する実行可能イメージを拡張特権付きでインストールします。そのようなイメージ (実行可能イメージのみ) を実行するプロセスには強い特権が一時的に割り当てられ、イメージの実行中に利用者登録ファイル (UAF) で定義された特権の制限事項を超える動作ができます。したがって、通常の特権を持つユーザでも、通常より高い特権が必要なプログラムを実行できます。

特権付きでインストールされたイメージから共用可能イメージなどの他のイメージを起動する場合には、そのイメージを特権イメージにリンクしたり、LIB\$FIND_IMAGE_SYMBOL を使用して、次の条件を満たす必要があります。

- 共用可能イメージは、INSTALL を使用して既知イメージとしてインストールしなければならない。
- そのイメージを検索するために使用する論理名およびテーブル名は、エグゼクティブ・モードあるいはカーネル・モードで定義されなければならない。特に、標準のエグゼクティブ・モードで定義した LNM\$FILE_DEV は LNM\$SYSTEM にしか変換されない。また、プロセス・テーブル、ジョブ・テーブル、あるいはグループ・テーブル内の定義は認識されない。
- 拡張特権付きでインストールするイメージは、必ず Linker ユーティリティの修飾子 /NODEBUG および /NOTRACE でリンクする。

3.10.8.2 特権付き共用可能イメージ

特権付き共用可能イメージとは、内部 (エグゼクティブまたはカーネル) モードで実行するエントリ・ポイントが定義された共用可能イメージです。共用可能イメージの内部モードのエントリ・ポイントは、ユーザが作成したシステム・サービスとして扱われます。

特権付き共用可能イメージを作成する場合、次の作業が必要です。

1. PLV (特権付きライブラリ・ベクタ) データ構造の VECTOR 属性を持つプログラム・セクションを作成する。
2. コマンド修飾子 /PROTECT または Linker ユーティリティの PROTECT= オプションを使用して共用可能イメージをリンクし、そのイメージを拡張特権の特定の形式にする。
 - イメージのすべての部分を保護する必要がある場合は、コマンド修飾子 /PROTECT を使用する。
 - 特権付きの共用可能イメージの一部だけを保護する必要がある場合は、PROTECT= オプションを使用する。
3. Install ユーティリティで修飾子 /PROTECTED および /SHARED の両方を指定し、特権付き共用可能イメージをインストールする。/PROTECTED 修飾子は保護属性を、/SHARED 修飾子は共用属性をそれぞれ割り当てる。これらの属性については、3.10.3 項「既知イメージに割り当てることができる属性」を参照。



注意:

INSTALL コマンドの ADD または CREATE に /PRIVILEGED 修飾子を指定して共用可能イメージに特権を付与することはできません。この修飾子は、実行可能イメージの場合にのみ動作します。

特権付き共用可能イメージの作成についての詳細は、『OpenVMS Programming Concepts Manual』を参照してください。

3.10.9 特権コンテキストでのイメージの起動方法

プロセスが、次のいずれかの動作を行う場合は、イメージ・アクティベータが制限付きの操作モードに入ります。これは、特権プログラムが実行されている場合に操作モードに入ったときと同様です。

- 実行アクセスを持つが読み込みアクセスを持たない、実行可能イメージまたは共用可能イメージを実行する。
- /PRIVILEGE または /EXECUTE_ONLY 修飾子を指定してインストールされた実行可能イメージを起動する。
- 特権付き共用可能イメージによって呼び出されたイメージを起動する。

この操作モードには、次の制限があります。

- 実行専用イメージの実行中に起動されるすべての共用可能イメージがインストールされていないなければならない。
- イメージ・アクティベータは、イメージ・ファイルをオープンするときに**信用論理名** (エグゼクティブ・モードまたはカーネル・モードに関連する論理名のこと) しか使用しないように OpenVMS RMS に指示する。



注意:

実行専用の共用可能イメージを呼び出す実行可能イメージは、/EXECUTE_ONLY 修飾子を付けてインストールしなければなりません。この修飾子によって、実行可能イメージは、プロセスが実行権を持つが読み込み権を持たない共用可能イメージを起動できるようになります。

/EXECUTE_ONLY 修飾子は、実行可能イメージに対してのみ有効です。

この制限により、特権コンテキストで実行されている共用可能イメージが保護され、期待通りの動作が保証されます。

3.10.10 INSTALL によるファイル名の指定

INSTALL の各コマンドを使用する場合、そのファイル指定には既存の実行可能イメージあるいは共用可能イメージを指定する必要があります。OpenVMS のレコード管理サービス (RMS) は、各ファイル指定を次の省略時の値を使用して解決します。

- デバイスおよびディレクトリのタイプは SYS\$SYSTEM
- ファイル・タイプは .EXE

CREATE コマンドまたは REPLACE コマンドを使用して、特定のバージョンのファイルを既知のバージョンのファイルとして指定できます。同じファイルの他のバージョンが存在していても、該当するイメージの既知ファイルの検索時には、必ず指定したバージョンが一致します。

3.10.11 INSTALL によるイメージのインストール

この作業を始めるにあたって、次のことを理解してください。

- インストール済みイメージの属性 (3.10.3 項「既知イメージに割り当てることができる属性」を参照)。
- Install ユーティリティのファイル指定 (3.10.10 項「INSTALL によるファイル名の指定」を参照)。

作業方法

1. CMKRNL 特権を獲得する。

```
§ SET PROCESS/PRIVILEGES=CMKRNL
```

2. INSTALL を起動する。

```
§ INSTALL
```

3. CREATE コマンドを、次の形式で入力する。
CREATE ファイル名 [/ 修飾子...]

イメージに割り当てる属性に従って、次の修飾子のいずれかを指定する。

- /EXECUTE_ONLY
- /HEADER_RESIDENT
- /OPEN
- /PRIVILEGED
- /PROTECTED
- /RESIDENT (Alpha システムおよび I64 システムのみ)
- /SHARED
- /WRITABLE

イメージのインストール方法についての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の INSTALL コマンドの CREATE を参照してください。



注意:

Install ユーティリティそのものをインストールする場合には、その前にインストールしておかなければならない共用可能イメージがいくつかあります。そのような必須共用可能イメージ (SMG\$SHR, LIBOTS など) の中に使用できないものがあつた場合、Install ユーティリティは正常に実行できません。このような状況では INSTALL は機能しないため、その欠落したイメージをインストールすることもできません。この問題を解決するには、次のように INSTALL コマンドを再定義します。

```
§ DEFINE INSTALL SYS$SYSTEM:INSTALL.EXE;0
```

このようにして INSTALL コマンドを入力した場合、イメージ・アクティベータは INSTALL.EXE の既知のファイル・リストをチェックしないので、INSTALL コマンドは完了し、必要な共用イメージをインストールすることができるようになります。

3.10.12 INSTALL による既知イメージの表示

既知イメージに関する情報を表示するためには、INSTALL の LIST コマンドを使用します。

LIST コマンドの /FULL 修飾子を使って表示された情報を利用して、イメージのインストールが適切であるかどうかを判断することができます。

作業方法

1. INSTALL を起動する。

```
§ INSTALL
```

2. すべての既知イメージとその属性を表示するには、LIST コマンドを入力する。特定のイメージの属性を表示するには、次の形式でイメージの名前を指定する。

```
LIST ファイル名
```

次に例を示す。

```
INSTALL> LIST LOGINOUT
```

- 特定のイメージに関するすべての項目、すなわち、アクセス数、同時アクセス数、および作成されたグローバル・セクション数を表示する場合には、次のように、/FULL 修飾子を指定する。

LIST/FULL ファイル指定

例

LOGINOUT.EXE というインストール済みイメージのすべての情報を表示します。表示される項目は、アクセス数、同時アクセス数、および作成されたグローバル・セクションの数です。

```
$ INSTALL
INSTALL> LIST/FULL LOGINOUT
DISK$VMS551:<SYS2.SYSCOMMON.SYSEXE>.EXE
  LOGINOUT;2      Open Hdr Shar Prv
    Entry access count      = 36366
    Current / Maximum shared = 1 / 10
    Global section count    = 3
    Privileges = CMKRNL SYSNAM LOG_IO ALTPRI TMPMBX SYSPRV
INSTALL>
```

3.10.13 共用可能イメージ・ファイルの論理名の定義

共用可能イメージが SYS\$SHARE に存在しない場合、それに対してリンクされている実行可能イメージを実行するために、そのイメージの論理名を指定する必要があります。たとえば、STATSHR のファイル指定が SYS\$SHARE:STATSHR.EXE であれば論理名は不要です。しかし、STATSHR を SYS\$DEVICE:[TEST] に格納した場合、STATSHR を呼び出す実行可能イメージを実行する前に、論理名として STATSHR を定義する必要があります。論理名は、共用可能イメージをリンクしたときにその入力ファイル指定で使用した名前と同じもの (インストールで使用した名前とも同じ) を使用します。例を示します。

```
$ DEFINE STATSHR SYS$SYSDEVICE:[TEST] STATSHR
```

共用可能イメージの論理名を再定義すれば、実行可能イメージを呼び出して再リンクすることなく、その共用可能イメージを別のものに置き換えることができます。たとえば、次の文はファイル名 STATSHR を再定義しています。この名前は、STATSHR を呼び出す実行可能イメージのための、共用可能イメージ SYS\$SYSDEVICE:[MAIN]STATSHR.EXE の論理名になります。

```
$ DEFINE STATSHR SYS$SYSDEVICE:[MAIN] STATSHR
```



注意:

特権付きの実行可能イメージを実行する場合、プロセス論理名テーブルおよびグループ論理名テーブルで定義された論理名は無視されます。イメージの検索には、エグゼクティブ・モードまたはカーネル・モードで定義された論理名およびテーブル名だけが使用されます。

3.10.14 既知イメージの削除

INSTALL コマンドの REMOVE は、あるイメージの既知ファイル・エントリを削除した後、そのイメージのインストール時に作成されたグローバル・セクションをすべて削除します。ボリュームに対応する既知ファイル・エントリが存在する場合には、そのボリュームはディスマウントできないことに注意してください。ボリュームをディスマウントするには、そのボリュームに対応する既知イメージをすべて削除してください。また、それらのイメージを使用するプロセスがすべて終了するまで待つ必要があります。ファイルの状態を確認するには、DCL コマンドの SHOW DEVICES/FILES を使用します。

INSTALL の DELETE コマンドについては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の INSTALL の節を参照してください。

3.11 予約メモリ・レジストリ

SYSMAN ユーティリティからインタフェース経由で予約メモリ・レジストリを使用することにより、OpenVMS Alpha システムまたは I64 システムに、メモリ常駐セクション内で他の特権アプリケーションで使用するために取り置かれている大容量のメモリを設定することができます。さらに、予約メモリ・レジストリを使用することにより、割り当て済みの予約メモリを考慮しながら、AUTOGEN ユーティリティによって OpenVMS システムを適切にチューニングすることもできます。

予約メモリ・レジストリを使用すると、次の処理を行うことができます。

- メモリ常駐グローバル・セクション・オプションのために、システム非流動メモリを予約する。
- メモリ常駐グローバル・セクション割り当てオプションのために、システム非流動メモリの他に、適切な物理ページを予約する。

予約メモリ・レジストリには、システムのブート中、割り当て済みページがゼロになるように指定する機能があります。このオプションを使用すると、メモリ常駐グローバル・デマンド・ゼロ・セクションを作成するために必要な時間が短縮されます。

予約メモリ・レジストリのもう 1 つのオプションとして、予約メモリのメモリ常駐グローバル・セクションにマップするために必要なページ・テーブルのサイズに関するものがあります。このオプションが指定され、予約メモリがメモリ常駐グローバル・セクションで使用されると、メモリ常駐グローバル・セクションは、共用ページ・テーブルを使用して作成されます。

3.11.1 予約メモリ・レジストリの使用

OpenVMS には、メモリ常駐グローバル・デマンド・ゼロ・セクション内で使用するために、非流動メモリを予約するメカニズムがあります。予約メモリのメモリには、単純にシステムの非流動メモリ・サイズから差し引かれたものが使用されるか、またはページが事前に割り当てられます。

予約メモリ・レジストリを使用すると、AUTOGEN により、メモリ常駐セクション・ページをシステムの流動ページの計算に含まないように、システムが適切にチューニングされるようになります。AUTOGEN はシステムのページ・ファイル、プロセスの数、およびワーキング・セットの最大サイズを、システムの流動ページ・カウントに基づいて計算します。AUTOGEN が、別の目的で使用するために永久に予約されている物理メモリを考慮しない流動ページ・カウントに基づいてパラメータを調整すると、システムに重大な性能上の問題が発生することがあります。

さらに、予約メモリ・レジストリを使用すると、割り当てオプションが使用された場合に、隣接するアラインされたメモリをメモリ常駐セクションで使用できるようになります。



注意:

このセクションでは、予約メモリをグローバル・セクションで使用方法について説明していますが、この機能は、他の特権アプリケーションでも使用することができます。

3.11.1.1 予約メモリ・レジストリ・データ・ファイル

予約済みの非流動メモリを消費することにより、メモリの特性が、システムの初期化中 (ブート時) に読み込まれるデータ・ファイルに入力されます。データ・ファイルの操作の仕組みは、SYS\$LOADABLE_IMAGES:VMS\$SYSTEM_IMAGES.DATA と似ています (インストール固有のエグゼクティブ・ロードされたイメージを示します)。

このファイルの名前は、次のとおりです。

```
SYS$SYSTEM:VMS$RESERVED_MEMORY.DATA
```

このファイルは、SYSMAN ユーティリティによって (エグゼクティブ・ロードされたイメージのデータ・ファイルの場合と同様に) 保守されます。

3.11.1.2 AUTOGEN

予約メモリ・レジストリ・ファイル VMS\$RESERVED_MEMORY.DATA は、AUTOGEN フィードバック・メカニズムによって読み込まれ、システムの流動ページ・カウンタの設定が考慮されています。AUTOGEN は、システムのページ・ファイル、プロセスの数、およびワーキング・セットの最大サイズを、システムの流動ページ・カウンタに基づいて計算します。

3.11.1.3 予約メモリ・レジストリへのエントリの追加

データ・ファイルへのエントリの追加には、SYSMAN ユーティリティを使用します。そのため SYSMAN コマンドは、次のとおりです。

```
SYSMAN RESERVED_MEMORY ADD gs_name -  
    /GROUP = n -  
    /SIZE = {size of reserved memory, unit: MB} -  
    /[NO] ALLOCATE -  
    /[NO] ZERO -  
    /[NO] PAGE_TABLES
```

- `gs_name` フィールドは、この予約メモリに関連付けられたメモリ常駐グローバル・セクションの名前です。名前を 1 つ指定しなければなりません。
- `/GROUP` 修飾子がない場合、予約メモリの対象は、システム・グローバル・セクション (SYSGBL) になります。
- `/GROUP` 修飾子がある場合、予約メモリの対象は、グループ・グローバル・セクションになります。プロセスの UIC グループ番号 (8 進数) という値により、グループ・グローバル・セクションが作成されます。作成者の UIC グループ番号のプロセスだけが、このグローバル・セクションにアクセスすることができます。たとえば、UIC が [6,100] のプロセスがグループ・グローバル・セクションの作成者である場合、`/GROUP` 修飾子に対して指定されるグループ番号は 6 になります。
- `/ALLOCATE` 修飾子が指定されていない場合、または `/NOALLOCATE` 修飾子が指定されている場合、予約メモリは、システムの次のリブート中には割り当てられません。予約メモリは、システムの流動ページ・カウンタからのみ差し引かれ、メモリ常駐グローバル・セクションが作成された結果、フォルト・オプションが使用されることになります。
- `/ALLOCATE` 修飾子が指定された場合、システムの次のリブート中にシステムが割り当てられます。割り当てられたメモリは、システムの流動ページ・カウンタから差し引かれ、メモリ常駐グローバル・セクションが作成された結果、割り当てオプションが使用されることになります。ページの物理アラインメントは、予約メモリのサイズがわかっている場合にページをマップするために使用できる最大粒度ヒント係数に基づいています。可能な粒度ヒント係数は 512 ページ (4 MB) および 64 ページ (512 KB) です。したがって、システム・ページ・サイズが 8 KB とすると、予約メモリは、次のように物理的にアラインされます。

1. `size >= 4 MB`: physically aligned on a 4 Mbyte boundary

2. `size < 4 MB`: physically aligned on a 512 KB boundary

- `/ZERO` 修飾子が指定されていないか、または `/NOZERO` が指定されている場合、割り当て済みページは、システム初期化中にゼロになりません。これらのページは、グローバル・セクションが作成されるとゼロになります。
- `/ZERO` 修飾子を使用できるのは、`/ALLOCATE` 修飾子が指定されている場合に限られます。`/ZERO` 修飾子が指定されている場合、割り当て済みページは、システム初期化中にゼロになります。ゼロになったページは、メモリ常駐グローバル・セクションで必要です。ただし、これらのページは、システム初期化中にゼロにする必要はありません。
- `/PAGE_TABLES` 修飾子が指定されていないか、または `/NOPAGE_TABLES` が指定されている場合、共用ページ・テーブルのための追加のメモリが取り置かれることはありません。メモリ常駐グローバル・セクションが作成されても、このグローバル・セクションのための共用ページ・テーブルが作成されることはありません。

- /PAGE_TABLES 修飾子が指定されている場合、共用ページ・テーブルのための追加のメモリが取り置かれます。メモリ常駐グローバル・セクションが作成されると、このグローバル・セクションのための共用ページ・テーブルが作成されます。/ALLOCATE 修飾子が指定されていないか、または/NOALLOCATEが指定されている場合、追加の予約メモリは、システムの流動ページ・カウントから差し引かれるだけです。/ALLOCATE 修飾子が指定されている場合、共有ページ・テーブルのためにシステムの次のリブート中に追加のページが割り当てられ、追加の予約メモリがシステムの流動ページ・カウントから差し引かれます。

3.11.2 予約メモリ・レジストリからのエントリの削除

SYSMAN コマンドを次の形式で実行することで、予約メモリ・エントリを削除することができます。

```
SYSMAN> RESERVED_MEMORY REMOVE gs_name /GROUP = n
```

指定される `gs_name` は、予約メモリ・レジストリから削除されるエントリに関連付けられたメモリ常駐セグションの名前です。名前を 1 つ指定しなければなりません。

/GROUP 修飾子によって指定される値 `n` は、削除されるメモリ常駐セグションに関連付けられた UIC グループ番号 (8 進数) です。メモリ常駐グローバル・セクションがグループ・セグションの場合、/GROUP 修飾子を指定しなければなりません。メモリ常駐グローバル・セグションがシステム・グローバル・セグションの場合、/GROUP 修飾子は指定しないでください。

ページ・テーブルが指定されたメモリ常駐グローバル・セグションのために取り置かれている場合、追加の予約メモリも削除されます。

REMOVE コマンドによって削除されるのは、予約メモリ・レジストリ・データ・ファイルのエントリだけです。実行中のシステムのメモリが影響を受けることはありません。

3.11.2.1 予約メモリの割り当て

システム初期化中に、VMS\$RESERVED_MEMORY.DATA データ・ファイルが読み込まれます。

このデータ・ファイルにあるエントリごとに、そのサイズが M バイト (MB) 単位で、RESERVED_MEMORY ADD コマンドで /SIZE 修飾子によって指定されたこのメモリ常駐グローバル・セグションのために、システムの流動ページ・カウントから差し引かれます。/PAGE_TABLES が指定された場合、このメモリ常駐グローバル・セグションをマッピングする共用ページ・テーブルに必要な量のメモリも、システムの流動ページ・カウントから差し引かれます。

/ALLOCATE が RESERVED_MEMORY ADD コマンドで指定された場合、メモリ常駐グローバル・セグションのために、適切な大きさの物理ページも割り当てられ、取り置かれます。/PAGE_TABLES が指定された場合、共用ページ・テーブルのために、適切な大きさの物理ページが割り当てられ、取り置かれます。これらのページには、指定されたサイズに対応する、最大粒度ヒント係数に適した物理アラインメントがあります。/ZERO が指定された場合、それらのページは、システム初期化中にまたはシステムの空き時間にゼロになります。/ZERO が指定されなかった場合、または /NOZERO が指定された場合、それらのページは、メモリ常駐グローバル・セグションが作成されるとゼロになります。

システム・パラメータ STARTUP_P1 が MIN に設定されると、予約メモリ・レジストリのエントリは無視され、メモリは予約されません。

システム流動ページを予約するか、隣接するアラインされた物理ページを割り当てて予約メモリ・レジストリ・データ・ファイルを処理中、システム初期化中にエラーが発生すると、エラー・メッセージがコンソールに表示され、システムはブートを続けます。

3.11.2.2 予約メモリの解放

実行中のシステムで、次の形式で SYSMAN コマンドを入力することで、予約メモリを解放することができます。

```
SYSMAN> RESERVED_MEMORY FREE gs_name /GROUP = n
```

指定される `gs_name` は、予約メモリ・レジストリから解放されるエントリに関連付けられたメモリ常駐セクションの名前です。名前を 1 つ指定しなければなりません。

`/GROUP` 修飾子によって指定された値 `n` は、解放されるメモリ常駐セクションに関連付けられた UIC グループ番号 (8 進法) です。メモリ常駐グローバル・セクションがグループ・グローバル・セクションの場合、`/GROUP` 修飾子を指定しなければなりません。メモリ常駐グローバル・セクションがシステム・グローバル・セクションの場合、`/GROUP` 修飾子を指定しないでください。

このグローバル・セクションのために、システム初期化中に物理ページが割り当て済みでなかった場合、予約メモリは単純に、システムの流動ページ・カウントに追加されるだけです。それ以外の場合、物理ページは、システムの未使用ページ・リストまたはゼロ・ページ・リストに割り当て解除されます。システムの流動ページ・カウントは、この割り当て解除されるページが含まれるように調整されます。

指定されたメモリ常駐グローバル・セクションのためにページ・テーブルも予約されている場合、共用ページ・テーブルのための予約メモリも解放されます。

予約メモリが、指定されたメモリ常駐グローバル・セクションによって使用中の場合、現在使用中でない予約メモリの量は解放されます。

`RESERVED_MEMORY FREE` コマンドは、予約メモリ・レジストリ・データ・ファイルの内容には影響を与えず、実行中のシステム内のメモリにのみ影響を与えます。

3.11.2.3 予約メモリの表示

予約メモリ情報は、異なる 2 か所、つまり予約メモリ・レジストリ・データ・ファイルと、このデータ・ファイルのエントリに基づいてシステムの初期化中に作成された、実行中のシステムの予約メモリ・レジストリにあります。

予約メモリに関する情報がどこにあるかによって、使用する表示メカニズムも異なります。

実行中のシステムにある予約メモリ・レジストリを表示するメカニズムには、`SYSMAN`、`DCL SHOW MEMORY` コマンド、および `SDA` の 3 つがあります。

- `SYSMAN`

実行中のシステムにある予約メモリ・レジストリは、次の形式で `SYSMAN` コマンドを実行することによって表示することができます。

```
SYSMAN> RESERVED_MEMORY SHOW gs_name /GROUP = n
```

指定される `gs_name` は、実行中のシステムにある、表示対象のエントリに関連付けられたメモリ常駐グローバル・セクションの名前です。`gs_name` が指定されない場合、すべての登録されたグローバル・セクションのための予約メモリが表示されます。

`/GROUP` 修飾子によって指定される値は、表示対象のメモリ常駐グローバル・セクションに関連付けられた UIC グループ番号 `n` (8 進法) です。メモリ常駐グローバル・セクションがグループ・グローバル・セクションの場合、`/GROUP` 修飾子は指定しなければなりません。メモリ常駐グローバル・セクションがシステム・グローバル・セクションの場合、`/GROUP` 修飾子は指定しないでください。`/GROUP` 修飾子は、`gs_name` が指定されている場合に限って使用できます。

- `DCL SHOW MEMORY` コマンド

実行中のシステムにある予約メモリ・レジストリは、次の `DCL SHOW MEMORY` コマンドを実行することによって表示することができます。このコマンドを実行すると、予約メモリ/レジストリに関する情報を含む、実行中のシステムに関するすべてのメモリ常駐情報が表示されます。

`SHOW MEMORY /RESERVED` によって表示されるのは、実行中のシステムの予約メモリ・レジストリに関する情報だけです。

`SHOW MEMORY` によって表示される情報には、指定されたグローバル・セクションによって現在使用中のメモリの量も含まれます。さらに、ページ・テーブルのために予約されているメモリおよび (もしあれば) 現在使用中のメモリの量も表示されます。

- SDA

SDAにも、実行中のシステムの予約メモリ・レジストリの他、クラッシュ・ダンプ内の予約メモリ・レジストリについても表示するための、さまざまな拡張機能が含まれています。コマンド・インタフェースは、SDA SHOW MEMORY コマンドです (『HP OpenVMS System Analysis Tools Manual』を参照してください)。

3.11.2.4 予約メモリの使用

システム・サービス SYS\$CREATE_GDZRO および SYS\$CRMPSC_GDZRO_64 は、内部カーネル・モード OpenVMS Alpha ルーチンまたは I64 ルーチン呼び出して、予約メモリ・レジストリに登録されている予約メモリを使用します。

グローバル・セクションは、予約メモリ・レジストリに登録する必要がありません。グローバル・セクション名が予約メモリ・レジストリに登録されている場合でも、グローバル・セクションのサイズは、予約メモリのサイズと正確に一致しません。グローバル・セクションが登録されていない場合、またはグローバル・セクションが予約メモリ・レジストリに登録されていても、/NOALLOCATE が指定されている場合、メモリ常駐グローバル DZRO セクションに、フォルト・オプションが使用されます。このサイズが予約メモリよりも大きい場合、システムに十分な流動ページがないと、メモリ常駐グローバル DZRO セクションを作成するシステム・サービス呼び出しは、失敗します。

グローバル・セクションが予約メモリ・レジストリに登録され、/ALLOCATE が指定された場合、メモリ常駐グローバル DZRO セクションには、割り当てオプションが使用されます。グローバル・セクションのサイズは、予約され、割り当てられたメモリのサイズ以下でないと、システム・サービスの呼び出しによって、エラー SS\$_MRES_PFN SMALL が戻されます。

3.11.2.5 予約メモリの復帰

メモリ常駐グローバル・セクションが削除されると、そのグローバル・セクションに使用されていた物理ページは、このグローバル・セクションに事前に割り当てられていなかった場合、未使用ページ・リストに割り当て解除されます。システムの流動ページ・カウントは、このグローバル・セクションの予約メモリ・レジストリに予約されていないページについてのみ、調整されます。

メモリ常駐グローバル・セクションが削除されると、そのグローバル・セクションに使用されていた物理ページは、このグローバル・セクションに事前に割り当てられていた場合、予約メモリ・レジストリに戻されます。これらの物理ページは未使用ページ・リストに割り当て解除されず、予約されたまま残ります。システムの流動ページ・カウントは、全く調整されません。

予約メモリは、SYSMAN ユーティリティの RESERVED_MEMORY FREE コマンドを使用した場合にのみ、実行中のシステムに解放できます。



注意:

パーマネント・グローバル・セクションは、SYS\$DGBLSC を呼び出し、グローバル・セクションへの最後の参照が実行された時点で削除されます。非パーマネント・グローバル・セクションは、単にグローバル・セクションへの最後の参照が実行された時点で削除されます。

3.11.3 アプリケーションの構成

メモリ常駐グローバル・セクションを使用する OpenVMS Alpha アプリケーションまたは I64 アプリケーションを構成するには、次のステップを実行します。

1. SYSMAN RESERVED_MEMORY ADD コマンドを実行して、予約メモリの使用が必要であることを指定する。
2. AUTOGEN を実行して、システムの流動ページ・カウントを適切に設定し、システムのページ・ファイル、プロセスの数、および最大ワーキング・セット・サイズが適切に設定されるようフィードバックする。

3. システムをリブートして、予約メモリがシステムの流動ページ・カウントから差し引かれ、隣接するアラインされたページが必要に応じて割り当てられ、ゼロにされるようにする。

第4章 ファイル・システムのデータ・キャッシュの管理

この章では、XFC (Alpha および I64) および Virtual I/O Cache (VIOC) を管理する方法について説明します。これらは、Files-11 ファイル・システムが ODS-2 および ODA-5 ボリュームのデータをキャッシュするときを使用します。

この章の内容

この章では、次の作業について説明します。

作業	参照箇所
クラスタ全体でのキャッシングの禁止	4.3 項 「クラスタ全体でのキャッシングの禁止」
キャッシングを禁止した状態でのボリュームのマウント	4.4 項 「キャッシングを禁止した状態でのボリュームのマウント」
Extended File Cache (Alpha および I64) の管理	4.5 項 「XFC の管理 (Alpha および I64)」
Virtual I/O Cache の管理	4.6 項 「Virtual I/O Cache の管理」

この章では、次の概念について説明します。

概念	参照箇所
キャッシング	4.1 項 「キャッシングについて」
ファイル・システムのデータ・キャッシュ	4.2 項 「ファイル・システムのデータ・キャッシュについて」

4.1 キャッシングについて

Files-11 ファイル・システムは、性能を向上させるために、**キャッシング** と呼ばれるテクニックを使用します。Files-11 ファイルは、ディスクから最近読み込んだり、ディスクに書き込んだりしたデータのコピーを、**キャッシュ** と呼ばれる、メモリの領域に保持しています。

アプリケーションがデータを読み込むときに、ファイル・システムはデータがキャッシュにあるかどうかをチェックします。データがキャッシュにない場合には、I/O を実行して、ディスクからデータを読み込みます。

キャッシングは、読み込み性能を向上させます。メモリ (キャッシュ) からデータを読み込む方が、ディスクから読み込むよりもはるかに速いためです。

ハードウェア I/O サブシステムおよび OpenVMS には、キャッシングのいくつかのレベルがあります。一般的に、キャッシングのレベルが多いほど、データにアクセスする応答時間が短くなります。

4.2 ファイル・システムのデータ・キャッシュについて

ODS-2 ボリュームと ODS-5 ボリュームの場合には、Files-11 ファイル・システムに複数のキャッシュがあります。ファイル・ヘッダなどファイルのメタデータのための**メタデータ**・

キャッシュと、ファイル・データのためのデータ・キャッシュです。 Files-11 は、次に示す 2 つのシステム全体のデータ・キャッシュのうち、いずれか 1 つを使用できます。

ファイル・システムの データ・キャッシュ	説明
Virtual I/O Cache (VIOC)	元からあるデータ・キャッシュで、VAX, Alpha, および I64 システムで使用できる。
Extended File Cache (XFC)	OpenVMS Alpha バージョン 7.3 以降と I64 システムで使用でき、OpenVMS Alpha および I64 システムでのみ使用できる。VIOC よりも優れた性能とより多くの機能を提供する。XFC は、OpenVMS Alpha バージョン 7.3 以降および I64 での省略時のキャッシュである。

この章では、データ・キャッシュを管理する方法について説明します。メタデータ・キャッシュの管理方法については、『OpenVMS Performance Management』を参照してください。RMS は、データ・キャッシングを実行できるローカル・バッファおよびグローバル・バッファを利用することに注意してください。省略時の設定では、このキャッシングは許可されていません。RMS のローカル・バッファおよびグローバル・バッファを操作すると、I/O 性能に影響を与える恐れがあるためです。RMS のローカル・バッファ およびグローバル・バッファの管理方法については、『Guide to OpenVMS File Applications』を参照してください。また、変更ページ・リストは、キャッシュの一種であるため、ページはプロセスのワーキング・セットに戻され、ディスクには出力されないことにも注意してください。

XFC および VIOC はともに、OpenVMS Cluster 内で一貫性のあるデータを保守する仮想ブロック・キャッシュです。これらは、データ・ファイルとイメージ・ファイルの両方をキャッシュします。データ・キャッシュは、ライトスルー・キャッシュです。アプリケーションがデータをファイルに書き込むと、データは直接ディスクに書き込まれます。アプリケーションは、ディスク I/O が完了し、データがディスクに書き込まれるまで、待たなければなりません。

OpenVMS Cluster では、さまざまなノードがさまざまなデータ・キャッシュを使用することができます。このため、混合アーキテクチャ・クラスタの場合には、XFC による利点が生じます。OpenVMS Alpha ノードおよび I64 ノードは、XFC または VIOC を使用することができます。OpenVMS VAX ノードは、4.5.6.2 項「複合アーキテクチャ OpenVMS Cluster での XFC の使用」で説明しているように、VIOC だけを使用することができます。

XFC は、I/O 性能を向上させ、VIOC では 使用不可能な次のような機能を持っています。

- 先読みキャッシング
- キャッシュの自動サイズ変更
- より大きい最大キャッシュ・サイズ
- キャッシュできる閉じたファイル数 (無制限)
- キャッシュできる I/O の最大サイズの制御
- キャッシュ・メモリが静的か動的かの制御

4.3 クラスタ全体でのキャッシングの禁止

システム・スタートアップ時に、スタティック・システム・パラメータは、ファイル・システムがデータ・キャッシュを使用するかどうか、使用する場合には、どちらのデータ・キャッシュ (XFC または VIOC) を使用するかを制御します。システム・パラメータは、次の表に示すようにオペレーティング・システムによって異なります。

オペレーティング・システム	システム・パラメータ	使用可能	使用不可
OpenVMS Alpha および I64	VCC_FLAGS	1 または 2 (省略時の設定) ¹	0
OpenVMS VAX	VBN_CACHE_S	1 (省略時の設定)	0

¹ 1 で VIOC を選択、2 (省略時の設定) で XFC を選択します。

OpenVMS Cluster では、ファイル・システムのデータ・キャッシングを1つのノードで使用不可に設定すると、クラスタ全体でも使用不可になります。クラスタの他のノードは、そのノードがクラスタから離脱するか、VCC_FLAGS または VBN_CACHE_S を 0 以外の値に設定してリブートしないと、XFC または VIOC を使用できません。DCL コマンド SHOW MEMORY を使用すると、キャッシングが使用可能になっているかどうかを確認することができます。

クラスタ全体でキャッシングを禁止するには、OpenVMS Cluster のいずれかのノードで次の手順を実行します。

1. MODPARAMS.DAT を使用して、対応するシステム・パラメータ (VCC_FLAGS または VBN_CACHE_S) を 0 に設定する。
2. AUTOGEN を実行して、他のシステム・パラメータもこの新しい値を使用できるようにする。これは必須の作業ではないが、実行しておくことが望ましい。
3. 新しい値を有効にするには、システムをリブートしなければならない。

4.4 キャッシングを禁止した状態でのボリュームのマウント

ファイル・システムがデータをデータベース・ボリュームなど特定のボリュームにキャッシュするのを禁止するには、MOUNT /NOCACHE コマンドを使用して、キャッシングを禁止した状態でボリュームをマウントします。

OpenVMS Cluster で XFC を使用している場合には、ボリュームを /NOCACHE でマウントした方が、SET FILE /CACHING_ATTRIBUTE を使用し、ボリューム内のすべてのファイルのキャッシング属性を、キャッシングしないように設定するよりも簡単です (4.5.4 項「ファイルのキャッシングの禁止」を参照してください)。MOUNT /NOCACHE を使用すると、キャッシングのオーバーヘッドが最少になります。MOUNT/NOCACHE コマンドを使用すると、XQP キャッシングも使用不可能になることに注意してください。

例

この例では、ORACLE_VOL1 というラベルのデータベース・ボリュームを、キャッシングを禁止してマウントしています。

```
$ MOUNT DUA100: ORACLE_VOL1 /NOCACHE /SYSTEM
```

4.5 XFC の管理 (Alpha および I64)

この節では、OpenVMS Alpha および I64 システムでのみ使用可能な XFC を管理する方法について説明します。次の作業について説明します。

作業	参照箇所
XFC と古いバージョンとの相互動作の保証	4.5.1 項「XFC と古いバージョンとの相互動作の保証」
キャッシュのサイズの管理	4.5.2 項「キャッシュのサイズの管理」
最大キャッシュ I/O サイズの管理	4.5.3 項「最大キャッシュ I/O サイズの管理」
ファイルのキャッシングの禁止	4.5.4 項「ファイルのキャッシングの禁止」
先読みキャッシングの禁止	4.5.5 項「先読みキャッシングの禁止」
性能の監視	4.5.6 項「性能の監視」
複合アーキテクチャ OpenVMS Cluster での XFC の使用	4.5.6.2 項「複合アーキテクチャ OpenVMS Cluster での XFC の使用」

4.5.1 XFC と古いバージョンとの相互動作の保証

以前のバージョンの OpenVMS Alpha または OpenVMS VAX を含む OpenVMS Cluster システムで、OpenVMS バージョン 7.3 以降または 164 システムの XFC を使用する場合は、以前のバージョンの OpenVMS を実行しているシステム上に、修正キットをインストールしなければなりません。



重要:

上記の修正キットは、XFC の前身である VIOC のキャッシュ・ロック・プロトコルの誤りを修正し、古いバージョンのキャッシュが新しい XFC と正しく相互動作するようにします。修正キットのこの修正を盛り込まないと、システムやプロセスがハングアップすることがあります。

4.5.2 キャッシュのサイズの管理

この節では、XFC の最小サイズおよび最大サイズを管理する方法について説明します。

XFC は、S2 空間の仮想メモリに保持され、I/O 負荷およびシステムで使用可能な共用メモリ量によって、自動的に縮小したり拡張したりします。S2 空間は 64 ビットのアドレス空間であるため、必要な場合には、キャッシュを非常に大きいサイズまで拡張することができます。I/O 負荷が増えるに連れてキャッシュも自動的に拡張しますが、最大サイズより大きくなることはありません。アプリケーションでメモリが必要になると、キャッシュは自動的に縮小しますが、最小サイズより小さくなることはありません。

4.5.2.1 最小キャッシュ・サイズの制御

XFC の最小サイズは、予約メモリ・レジストリの VCC\$MIN_CACHE_SIZE エントリの値によって制御します。VCC\$MIN_CACHE_SIZE では、システム・スタートアップ時に XFC に割り当てられたメモリの量を、M バイト (MB) 単位で指定します。キャッシュは、このサイズを超えて縮小することはありません。このメモリが解放されることは絶対にありません。

最小サイズのチェック

XFC の最小サイズをチェックするには、Sysman ユーティリティ・コマンド RESERVED_MEMORY /SHOW または DCL コマンド SHOW MEMORY /RESERVED を使用します。次に例を示します。

```
$ SHOW MEMORY /RESERVED
      System Memory Resources on 11-MAY-2000 15:50:25.64

Memory Reservations (pages):      Group      Reserved      In Use      Type Allocated
      VCC$MIN_CACHE_SIZE           ---           1536         1536
      Total (400.00 Mb reserved)    1536         1536
```

最小サイズの設定

省略時の設定では、予約メモリ・レジストリには VCC\$MIN_CACHE_SIZE のエントリが含まれていないため、システム・スタートアップ時に XFC に割り当てられるメモリはありません。ただし、XFC は、システム全体のスループットを保守するために、ごくわずかの量のメモリを割り当てます。このとき割り当てられるメモリの量は、使用しているコンピュータのサイズによって異なります。

最小サイズを設定するには、次の手順に従ってください。

1. Sysman ユーティリティの RESERVED_MEMORY ADD コマンドを使用して、VCC\$MIN_CACHE_SIZE のエントリを追加する。最小サイズを 300 MB に設定する例を次に示す。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> RESERVED_MEMORY ADD VCC$MIN_CACHE_SIZE /SIZE=300 /ALLOCATE -
  _SYSMAN> /NOGLOBAL_SECTION /NOZERO /NOPAGE_TABLE
```

この例で示されている修飾子は、すべて使用しなければならない。最大限の性能を実現するには、最小サイズを 4 MB の倍数に設定する。

この作業を、予約メモリを異なった RAD に割り当てようとしている NUMA タイプのコンピュータで実行している場合に最小サイズを設定するには、(先の例のコマンドではなく) 次のコマンドを使用する。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> RESERVED_MEMORY ADD VCC$MIN_CACHE_SIZE /SIZE=300 /ALLOCATE -
SYSMAN> /NOGLOBAL_SECTION /NOZERO /NOPAGE_TABLE/RAD=0
SYSMAN> RESERVED_MEMORY EXTEND VCC$MIN_CACHE_SIZE /SIZE=500 /ALLOCATE -
SYSMAN> /NOGLOBAL_SECTION /NOZERO /NOPAGE_TABLE/RAD=1
```

2. AUTOGEN を実行して、他のシステム・パラメータもこの新しい値を使用できるようにする。これは必須の作業ではないが、実行しておくことが望ましい。
3. システムをリブートして、新しい値を有効にする。

スタートアップ中に、システムに指定された最小サイズを割り当てる十分なメモリがない場合には、XFC にメモリは割り当てられず、最小サイズは 0 MB に設定される。

最小サイズの変更

XFC の最小サイズを変更するには、次の手順に従ってください。

1. Sysman ユーティリティの RESERVED_MEMORY MODIFY コマンドを使用して、VCC\$MIN_CACHE_SIZE の既存のエントリを変更する。最小サイズを 360 MB に変更する例を次に示す。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> RESERVED_MEMORY MODIFY VCC$MIN_CACHE_SIZE /SIZE=360 /ALLOCATE -
SYSMAN> /NOGLOBAL_SECTION /NOZERO
```

この例で示されている修飾子は、すべて使用しなければならない。最大限の性能を実現するには、最小サイズを 4 MB の倍数に設定する。

2. AUTOGEN を実行して、他のシステム・パラメータもこの新しい値を使用できるようにする。これは必須の作業ではないが、実行しておくことが望ましい。
3. 新しい値を有効にするには、システムをリブートしなければならない。

スタートアップ中に、システムに指定された最小サイズを割り当てる十分なメモリがない場合には、XFC にメモリは割り当てられず、最小サイズは 0 MB に設定される。

4.5.2.2 最大キャッシュ・サイズの制御

XFC の最大サイズを制御するには、ダイナミック・システム・パラメータ VCC_MAX_CACHE を使用します。サイズを M バイト単位で指定します。

省略時の設定では、VCC_MAX_CACHE は -1 で、これは、システム・スタートアップ時に、XFC の最大サイズがシステム上の物理メモリの 50 パーセントに設定されることを意味します。たとえば、システムの物理メモリが 2 GB である場合には、その最大サイズは 1 GB です。

VCC_MAX_CACHE によって指定される最大サイズには、XFC が OpenVMS ロック・マネージャを介して間接的に消費するメモリが含まれていないことに注意してください。

システム・スタートアップ時の VCC_MAX_CACHE の値によって、XFC の最大サイズが決まります。この値を超えて最大サイズを増やすことはできません。

たとえば、システム・スタートアップ時に VCC_MAX_CACHE の値が 60 である場合には、最大サイズは最初 60 MB に設定されます。その後、VCC_MAX_CACHE の値を 40 に設定すると、最大サイズは縮小し、40 MB になります。XFC のサイズが 40 MB より大きい場合には、そのサイズは徐々に縮小し、40 MB になります。次に VCC_MAX_CACHE の値を 80 に設定しても、最大サイズは、システム・スタートアップ時に設定された値である 60 MB までしか

拡張しません。システム・スタートアップ時に設定された値を超えて最大サイズを増やすことはできないためです。

VCC_MAX_CACHE が、予約メモリ・レジストリの VCC\$MIN_CACHE_SIZE エントリの値によって指定された最小サイズより小さい場合には、システム・スタートアップ時に VCC_MAX_CACHE は無視され、XFC は、この最小サイズと同じ値に設定されます。この場合には、XFC は固定サイズになり、縮小も拡大もできません。

例

アクティブなシステム上の XFC の最大サイズを、60 MB から 40 MB に減らす例を次に示します。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE ACTIVE
SYSGEN> SET VCC_MAX_CACHE 40
SYSGEN> WRITE ACTIVE
```

```
$ SET CACHE /RESET
```

現在のパラメータ・セットを変更することにより、リブートを行っても変更が有効なままとする例を次に示します。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE CURRENT
SYSGEN> SET VCC_MAX_CACHE 40
SYSGEN> WRITE CURRENT
```

この変更を永続的なものにするには、変更内容を MODPARAMS.DAT に入力しなければなりません。この変更は、次のリブート後に有効になります。

4.5.2.3 静的キャッシュ・サイズの許可

より大規模なコンピュータでは、XFC は変更ページ・リストの省略時のサイズによって、多少制限されることがあります。通常、変更ページ・リストは、1対1対応のデータ・キャッシュと考えられます。XFC は多対1対応のキャッシュであるため、通常、1つのキャッシュ・ページが多数のユーザによってアクセスされます。大規模なメモリ・システムでは、AUTOGEN は通常、MPW_HILIMIT を非常に大きい値に設定します。このため、メモリ・サブシステムが XFC に与えるための空きメモリが十分でないことも考えられます。

XFC の最小ページ数は、4.5.2.1 項「最小キャッシュ・サイズの制御」の最小サイズの設定で指定されたとおりに強制的に設定することができます。さらに、XFC 専用のメモリを永久的に割り当てて、動的な割り当てルーチンおよび割り当て解除ルーチンによってオーバーヘッドが発生するのを防ぐこともできます (VIOC に対する操作と似ています)。このようにするには、システム・パラメータ VCC_MAX_CACHE を、VCC\$MIN_CACHE_SIZE によって指定されるメモリ予約と等しくなるように設定します。

たとえば、システムのメモリが 128 GB ある場合には、8 GB のメモリを XFC に占有させることによって、キャッシュのヒット率および全体の応答時間は、常に良くなります。次に例を示します。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE CURRENT
SYSGEN> SET VCC_MAX_CACHE 8000
SYSGEN> WRITE CURRENT
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> RESERVED_MEMORY ADD VCC$MIN_CACHE_SIZE /SIZE=8000 /ALLOCATE -
_SYSMAN> /NOGLOBAL_SECTION /NOZERO /NOPAGE_TABLE/RAD=0
```

VCC_MAX_CACHE への変更を永続的なものにするには、変更内容を MODPARAMS.DAT に入力しなければなりません。この変更は、次のリブート後に有効になります。AUTOGEN を実行して、他のシステム・パラメータが新しい値を許すか確認することをお勧めします。

4.5.3 最大キャッシュ I/O サイズの管理

ダイナミック・システム・パラメータ `VCC_MAX_IO_SIZE` は、XFC でキャッシュできる I/O の最大サイズを制御します。このパラメータでは、サイズをブロック単位で指定します。省略時の値は 127 です。

例

この例では、XFC でキャッシュできる I/O の最大サイズを、1000 ブロックに変更します。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE ACTIVE
SYSGEN> SET VCC_MAX_IO_SIZE 1000
SYSGEN> WRITE ACTIVE
```

この一連のコマンドは、将来マウントされるボリュームだけでなく、現在ローカル・ノードにマウントされているボリュームへの I/O にも影響を与えます。これらのコマンドを入力した後、XFC は 1000 ブロックより大きい I/O をキャッシュしなくなります。SHOW MEMORY /CACHE /FULL コマンドを実行すると、効率的なパラメータの設定のためのガイドラインとなる、I/O サイズの棒グラフが表示されます。

4.5.4 ファイルのキャッシングの禁止

XFC が、データベース・ファイルなど特定のファイルをキャッシングするのを禁止するには、対象となるファイルのキャッシング属性を `NO_CACHING` に設定します。

ファイルの**キャッシング属性**とは、アプリケーションが使用するキャッシング・オプションを指定せずにファイルにアクセスした場合に、XFC が使用する省略時のキャッシング・オプションです。このキャッシング・オプションは、`WRITETHROUGH` または `NO_CACHING` です。

ファイルをキャッシュするには、そのキャッシング属性を `WRITETHROUGH` (省略時の設定) に設定します。ファイルをキャッシュしないようにするには、そのキャッシング属性を `NO_CACHING` に設定します。

属性	説明
<code>SET FILE /CACHING_ATTRIBUTE=keyword¹</code>	ファイルまたはディレクトリにキャッシング属性を設定する。
<code>DIRECTORY /CACHING_ATTRIBUTE</code> または <code>DIRECTORY /FULL</code>	ファイルまたはディレクトリのキャッシング属性を表示する。
<code>DIRECTORY /SELECT=CACHING_ATTRIBUTE=(keyword[,...])¹</code>	特定のキャッシング属性を持つすべてのファイルおよびディレクトリを表示する。

¹ キーワードは `WRITETHROUGH` または `NO_CACHING` のいずれか。

XFC はディレクトリをキャッシュしません。ディレクトリのキャッシング属性は、そのディレクトリ内に新たに作成されるファイルやサブディレクトリに、どのようにキャッシング属性が継承されるかのみを制御します。

- 新しいディレクトリまたはファイルを作成すると、親ディレクトリからキャッシング属性を継承する。
- 既存ファイルの新しいバージョンを作成すると、新しいバージョンは既存ファイルの最新バージョンからキャッシング属性を継承する。

例

1. この例では、`[SMITH.BORING]` ディレクトリおよびその下位ディレクトリ内のファイルすべてに対して、キャッシング属性を `NO_CACHING` に設定する。

最初の `SET FILE` コマンドは、ディレクトリ内にその後作成されたすべてのファイルおよびサブディレクトリがその属性を継承するように、ディレクトリに対して属性を設定する。

2 番目の SET FILE コマンドは、ディレクトリおよびその下位ディレクトリ内のすべての既存ファイルおよびディレクトリに対して属性を設定する。

```
$ SET FILE DISK$USERS:[SMITH]BORING.DIR;1 /CACHING_ATTRIBUTE=NO_CACHING
$ SET FILE DISK$USERS:[SMITH.BORING...]*.*;* /CACHING_ATTRIBUTE=NO_CACHING
```

2. この例では、DIRECTORY コマンドの /CACHING_ATTRIBUTE 修飾子を使用して、MYFILE.TXT のキャッシング属性を表示する。

```
$ DIRECTORY MYFILE.TXT /CACHING_ATTRIBUTE
Directory DISK$USERS:[SMITH]
```

```
MYFILE.TXT;1          Write-through
```

```
Total of 1 file.
```

3. この例では、DISK\$USERS ボリュームにある、キャッシング属性が NO_CACHING であるすべてのファイルを表示する。

```
$ DIRECTORY DISK$USERS:[000000...]*.* /SELECT=CACHING_ATTRIBUTE=NO_CACHING
```

4.5.5 先読みキャッシングの禁止

XFC は、データを順に読み込むアプリケーションの性能を向上させるために、**先読みキャッシング**と呼ばれるテクニックを使用します。ファイルがサイズの等しい I/O に順に読み込まれると、これを検出し、現在の読み込みよりも先にデータをフェッチして、次の読み込み命令がキャッシュから満たされるようにします。

ローカル・ノードでの先読みキャッシングを禁止するには、ダイナミック・システム・パラメータ VCC_READAHEAD を 0 に設定します。省略時の設定ではこのパラメータは 1 で、ローカル・ノードが先読みキャッシングを実行できます。

例

この例では、ローカル・ノードでの先読みキャッシングを禁止します。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> USE ACTIVE
SYSGEN> SET VCC_READAHEAD 0
SYSGEN> WRITE ACTIVE
```

この一連のコマンドは、将来マウントされるボリュームだけでなく、現在ローカル・ノードにマウントされているボリュームへの I/O にも影響を与えます。これらのコマンドを入力した後、先読みキャッシングはローカル・ノードでは使用されなくなります。

4.5.6 性能の監視

XFC は VIOC よりも多くの情報を提供します。たとえば、システム全体、ボリューム全体、またはファイル単位でも情報を取得することができます。ディスク I/O 応答時間も取得可能です。SHOW MEMORY コマンドについては、『OpenVMS DCL ディクショナリ: N-Z』を参照してください。

4.5.6.1 システム全体の統計情報

SHOW MEMORY /CACHE を使用すると、XFC のシステム全体での性能を監視することができます。次に例を示します。

```
$ SHOW MEMORY /CACHE
```

```
System Memory Resources on 26-JAN-2001 15:58:18.71
```

```
Extended File Cache (Time of last reset: 24-JAN-2001 15:03:39.05)
```

Allocated (Mbytes)	1	3000.00	Maximum size (Mbytes)	2	5120.00
Free (Mbytes)	3	2912.30	Minimum size (Mbytes)	4	3000.00
In use (Mbytes)	5	87.69	Percentage Read I/Os	6	98%
Read hit rate	7	92%	Write hit rate	8	0%
Read I/O count	9	178136	Write I/O count	10	1867
Read hit count	11	165470	Write hit count	12	0
Reads bypassing cache	13	2802	Writes bypassing cache	14	39
Files cached open	15	392	Files cached closed	16	384
Vols in Full XFC mode	17	0	Files in VIOC Compatible mode	18	4
Vols in No Caching mode	19	1	Vols in Perm. No Caching mode	20	0

1 Allocated

このキャッシュに現在割り当てられているメモリの量。

2 Maximum size

キャッシュが拡張できる最大サイズ。

3 Free

使用されていないキャッシュに現在割り当てられているメモリの量。

4 Minimum size

キャッシュが縮小できる最小サイズ。これは、予約メモリ・レジストリの VCC\$MIN_CACHE_SIZE エントリによって制御される。

5 In Use

使用されているキャッシュに現在割り当てられているメモリの量。これは、Allocated の値と Free の値の差に相当する。

6 Percentage Read I/Os

読み込まれる I/O のパーセント。

7 Read hit rate

Read hit count フィールドを Read I/O count フィールドで除算して生じる数値 (率)。

8 Write hit rate

このフィールドは、将来使用するために予約されている。

9 Read I/O count

システム・スタートアップ以降、キャッシュによって確認された読み込み I/O の総数。

10 Write I/O count

システム・スタートアップ以降、キャッシュによって確認された書き込み I/O の総数。

11 Read hit count

システム・スタートアップ以降の読み込みヒットの総数。読み込みヒットとは、データがキャッシュで見つかったため、ディスクへの物理 I/O が必要なかった読み込み I/O である。

12 Write hit count

このフィールドは、将来使用するために予約されている。

13 Reads bypassing cache

システム・スタートアップ以降、キャッシュによって確認されたが、キャッシュされなかった読み込み I/O の総数。その理由としては、大きすぎる /NOCACHE を使用してマウントされたボリュームに対するものである、QIO 修飾子 IO\$m_DATACHECK, IO\$m_INHRETRY, または IO\$m_NOVCACHE のいずれかを使用して指定されているなどが挙げられる。

14 Write bypassing cache

システム・スタートアップ以降、キャッシュによって確認されたが、キャッシュされなかった書き込み I/O の総数。その理由としては、大きすぎる /NOCACHE を使用してマウントされたボリュームに対するものである、QIO 修飾子 IO\$m_DATACHECK,

IO\$M_ERASE, IO\$M_INHRETRY, または IO\$M_NOVCACHE のいずれかを使用して指定されているなどが挙げられる。

- 15 Files cached open
現在キャッシュされているオープンされたファイルの数。
- 16 Files cached closed
有効なデータがまだキャッシュに入っている、閉じられたファイルの数。
- 17 Volumes in Full XFC mode
弊社で、将来使用するために予約済み。0 になる。
- 18 Volumes in VIOC compatible mode
VCC キャッシング・プロトコルを使用している XFC によってキャッシングされるボリュームの数。OpenVMS バージョン 7.3 以降、XFC が使用するのは VCC キャッシング・プロトコルだけである。
- 19 Volumes in No Caching mode
ローカル・ノードまたは OpenVMS Cluster の別のノードのいずれかでキャッシングが禁止されている場合に、この値は、ローカルノードに現在マウントされているボリュームの数になる。それ以外の場合は 0。
- 20 Vols in Perm. No Caching mode
このフィールドは 0 でなければならない。0 以外の数字である場合には、XFC がこのデバイスへの不正な書き込み操作を検出して、このデバイスへのキャッシングを禁止したことを意味する。

SHOW MEMORY コマンドについては、『OpenVMS DCL デイクシヨナリ: N-Z』を参照してください。

4.5.6.2 複合アーキテクチャ OpenVMS Cluster での XFC の使用

OpenVMS Cluster では、XFC を使用できるノードもあれば、VIOC を使用できるノードもあります。このため、混合アーキテクチャ・クラスタの場合には、XFC による利点が生じます。VIOC を使用しているノード上にボリュームがマウントされている場合には、XFC を使用しているノードは、書き込みのために共有されているボリュームにあるどのファイルについても、キャッシュできません。書き込みのために共有されているファイルとは、OpenVMS Cluster 内の複数のノードによってアクセスされているファイルで、これらのノードのうち少なくとも 1 つが、書き込みアクセスのために開いているファイルのことです。

4.6 Virtual I/O Cache の管理

この節では、VIOC の管理方法について説明します。次の作業について説明します。

作業	参照箇所
Alpha システムまたは I64 システムでの VIOC の選択	4.6.2 項「Alpha システムまたは I64 システムでの VIOC の選択」
キャッシュのサイズの管理	4.6.3 項「キャッシュのサイズの管理」
性能の監視	4.6.4 項「VIOC 統計情報の表示」

Virtual I/O Cache とは、クラスタ全体を対象にした、ライトスルーの、ファイル指向のディスク・キャッシュです。ディスク I/O 操作の数を低減し、性能を向上させることができます。Virtual I/O Cache の目的は、I/O 応答時間を最小限のオーバーヘッドに抑えることによって、システム・スループットを向上させることです。Virtual I/O Cache は、システム管理およびアプリケーション・ソフトウェアからは透過的に動作し、仮想ディスク I/O の読み込み性能を非常に向上させながら、システムの信頼性を維持します。

4.6.1 Virtual I/O Cache のしくみについて

Virtual I/O Cache は、データ・ファイルおよびイメージ・ファイルを格納することができます。たとえば、ODS-2 ディスク・ファイルのデータ・ブロックは、最初にアクセスされるときに Virtual I/O Cache にコピーされます。その後、同じデータ・ブロックの読み込み要求は、この Virtual I/O Cache で満たされ (ヒット)、本来必要な物理ディスク I/O の操作 (ミス) は不要になります。

システムの負荷によって、アプリケーションのスループットが向上したり、会話型応答が向上したり、I/O 負荷が低減したりします。



注意:

単一の読み込みおよび書き込み要求を開始するアプリケーションの場合には、データがキャッシュから再読み込みされることが全くないため、Virtual I/O Cache を使用しても利点はありません。暗黙に指定されている I/O 遅延時間に依存するアプリケーションは、強制終了したり、予想できない結果を生じたりする恐れがあります。

キャッシュがデータを操作する方法について、次のいくつかのポリシーが適用されます。

- ライトスルー — すべての書き込み I/O 要求は、ディスクと同様にキャッシュにも書き込まれる。
- Least Recently Used (LRU) — キャッシュがいっぱいになると、最も以前に使用されたデータが置き換えられる。
- キャッシュされたデータはファイルを閉じた後も保存 — データは、ファイルを閉じた後もキャッシュ内に残る。
- 読み込み要求や書き込み要求で割り当て — キャッシュ・ブロックは、読み込み要求や書き込み要求があると、割り当てられる。

4.6.2 Alpha システムまたは I64 システムでの VIOC の選択

何らかの理由により、Alpha システムまたは I64 システムで XFC ではなく VIOC を使用する必要がある場合には、次の手順に従います。

1. Sysman ユーティリティの RESERVED_MEMORY REMOVE コマンドを使用して、予約メモリ・レジストリから VCC\$MIN_CACHE_SIZE のエントリを削除する。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> RESERVED_MEMORY REMOVE VCC$MIN_CACHE_SIZE /NOGLOBAL_SECTION
```

これにより、ステップ 4 でシステムが VIOC を使用してリブートすると、XFC にメモリは割り当てられない。

2. VCC_FLAGS システム・パラメータを 1 に設定する。
3. AUTOGEN を実行して、他のシステム・パラメータもこの新しい値を使用できるようにする。これは必須の作業ではないが、実行しておくことが望ましい。
4. システムをリブートする。VCC_FLAGS が 1 であるため、スタートアップ中に、XFC ではなく VIOC が自動的にロードされる。

ステップ 1 で予約メモリ・レジストリから VCC\$MIN_CACHE_SIZE エントリを削除するのを忘れた場合には、XFC がロードされていないにもかかわらず、XFC にメモリが割り当てられません。このメモリは使用されません。このような状態になった場合には、Sysman ユーティリティの RESERVED_MEMORY FREE コマンドを使用して、このメモリを解放します。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> RESERVED_MEMORY FREE VCC$MIN_CACHE_SIZE /NOGLOBAL_SECTION
```

4.6.3 キャッシュのサイズの管理

VIOC のサイズを管理する方法は、使用しているシステムが OpenVMS Alpha か、I64 か、VAX かによって異なります。

OpenVMS Alpha システムまたは I64 システム

OpenVMS Alpha システムおよび I64 システムの場合には、VIOC のサイズは、システム・スタートアップ時に固定されます。キャッシュは縮小も拡張もできません。スタティック・システム・パラメータ VCC_MAXSIZE の値により、キャッシュのサイズがブロック単位で指定されます。省略時の値は、6400 ブロック (3.2 MB) です。

OpenVMS Alpha システムまたは I64 システムで VIOC のサイズを変更するには、次の手順に従います。

1. VCC_MAXSIZE システム・パラメータを必要な値に設定する。
2. AUTOGEN を実行して、他のシステム・パラメータもこの新しい値を使用できるようにする。これは必須の作業ではないが、実行しておくことが望ましい。
3. システムをリブートして、新しい値を有効にする。

OpenVMS VAX システム

OpenVMS VAX システムでは、スタティック・システム・パラメータ VCC_PTES を使用して、VIOC の最大サイズを指定することができます。このパラメータにより、サイズがページ単位で指定されます。省略時の値は 2,000,000,000 です。

使用しているシステムでの I/O 負荷および使用可能なメモリ量の残りに応じて、VIOC は自動的に縮小したり、拡張したりします。I/O 負荷が増加すると、キャッシュも自動的に拡張しますが、最大サイズを超えることはありません。アプリケーションでメモリが必要な場合には、キャッシュは自動的に縮小します。

OpenVMS VAX システムで VIOC のサイズを変更するには、次の手順に従います。

1. VCC_MAXSIZE システム・パラメータを必要な値に設定する。
2. AUTOGEN を実行して、他のシステム・パラメータもこの新しい値を使用できるようにする。これは必須の作業ではないが、実行しておくことが望ましい。
3. システムをリブートして、新しい値を有効にする。

4.6.4 VIOC 統計情報の表示

DCL コマンド SHOW MEMORY/CACHE/FULL を使用すると、Virtual I/O Cache に関する統計情報を表示することができます。次に例を示します。

```
§ SHOW MEMORY/CACHE/FULL
```

```
System Memory Resources on 10-OCT-2002 18:36:12.79
Virtual I/O Cache
Total Size (pages)      1  2422      Read IO Count          2  9577
Free Pages              3   18      Read Hit Count         4  5651
Pages in Use            5  2404      Read Hit Rate          6  59%
Maximum Size (SPTES)   7 11432      Write IO Count         8  2743
Files Retained          9   99      IO Bypassing the Cache 10  88
```



注意:

この例は、VAX システムでの SHOW MEMORY/CACHE/FULL コマンドの出力を示しています。Alpha システムまたは I64 システムでは、SHOW MEMORY/CACHE/FULL コマンドの出力は、多少異なって表示されます。

- 1 Total Size
VIOC が現在管理しているシステム・メモリ・ページの総数を表示する。
- 2 Read I/O Count
最新のシステム・ブート以降、VIOC によって確認されている読み込み I/O の総数を表示する。
- 3 Free Pages

VIOC が管理しているページのうち、キャッシュ・データが含まれていないページの数を表示する。

4 Read Hit Count

最新のシステム・ブート以降、データが確認されていないため、物理 I/O を実行していない読み込み I/O の総数を表示する。

5 Pages in Use

VIOC が管理しているページのうち、有効なキャッシュ・データが含まれているページの数を表示する。

6 Read Hit Rate

読み込みヒット数と読み込み I/O 数の率を表示する。

7 Maximum Size

キャッシュが拡張できる最大サイズを表示する。

8 Write I/O Count

最新のシステム・ブート以降、VIOC によって確認されている書き込み I/O の総数を表示する。

9 Files Retained

有効なデータがキャッシュ内に残っているため、ファイルを閉じた後もファイル・システムの管理情報が残っている、閉じられたファイルの数を表示する。

10 I/O Bypassing

何らかの理由で、キャッシュによる要求/更新を満たそうとしなかった I/O の数を表示する。

4.6.5 VIOC の許可

省略時の設定では、Virtual I/O Cache によるキャッシングは許可されています。キャッシングを許可したり禁止するには、次のシステム・パラメータを使用します。MODPARAMS.DAT のパラメータの値を、次のように変更します。

パラメータ	許可	禁止
VCC_FLAGS (Alpha または I64)	1	0
VBN_CACHE_S (VAX)	1	0

MODPARAMS.DAT を更新して、必要なパラメータの値を変更した後、AUTOGEN を実行して、キャッシングを許可または禁止する 1 つまたは複数のノードをリブートしなければなりません。キャッシングは、システムの初期化中に自動的に許可または禁止されます。ユーザによる操作は不要です。

4.6.6 VIOC が許可されているかどうかの確認

SHOW MEMORY/CACHE を使用すると、実行中のシステムで VIOC キャッシングがオンであるかオフであるかが表示されます。(SYSGEN を使用するよりもはるかに簡単です。)

SYSGEN を使用すると、システムがブートされる前にパラメータを調べることができます。たとえば、システム・パラメータ VCC_FLAGS (Alpha または I64) または VBN_CACHE_S (VAX) をチェックすると、SYSGEN を使用することにより、Virtual I/O Cache によるキャッシングが許可されているかどうかを知ることができます。次に Alpha システムの例を示します。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> SHOW VCC_FLAGS
```

値が 0 の場合には、キャッシングが禁止されています。値が 1 の場合には、キャッシングが許可されています。

4.6.7 メモリの割り当てと VIOC

キャッシングに割り当てられるメモリは、未使用ページ・リストのサイズによって決まります。次の条件のうちいずれか 1 つがあてはまる場合には、Virtual I/O Cache のサイズは、拡張することができます。

- 使用可能な未使用メモリの量が FREEGOAL の値の 2 倍あり、定期的にウェイクするプロセスで積極的なメモリ再生が許可されている場合
- 使用可能な未使用メモリの量が FREEGOAL の値に等しく、長時間待機しているプロセスで積極的なメモリ再生が許可されている場合
- 使用可能な未使用メモリの量が GROWLIM よりも多く、積極的なメモリ再生が許可されていない場合

キャッシュ・サイズは、次のような条件によっても制限されます。

- 使用可能なシステム・ページ・テーブル・エントリ (SPTE) の数。この数は、ブート時に決定された値によって計算される。
- メモリ管理サブシステムの需要。メモリ管理サブシステムはキャッシュに対する直接インタフェースを持っているため、必要に応じ、領域を戻すようにキャッシュに対して要求することができる。

キャッシュからのメモリ再生はどのように行われるのでしょうか。スワップは、第 1 レベルのトリミングを使用して、Virtual I/O Cache に割り当てられているメモリを再生することができます。さらに、ヒューリスティック・プリミティブがキャッシュを縮小し、メモリを小規模な増分単位で戻します。

4.6.8 VIOC サイズの調整

Virtual I/O Cache のサイズは、システム・パラメータ VCC_MAXSIZE によって決まります。このパラメータによって指定されるメモリの量がシステムの初期化時に静的に割り当てられ、Virtual I/O Cache によって引き続き所有されます。

キャッシュのサイズを増やしたり減らすには、VCC_MAXSIZE を変更し、システムをリブートします。

4.6.9 VIOC および OpenVMS Cluster の構成

キャッシュは、単一ノード・システムから大規模な混合インターコネクト OpenVMS Cluster システムにいたるまで、サポートされているすべての構成で動作します。Virtual I/O Cache は、中心点に相当します。つまり、キャッシュはそれぞれの OpenVMS Cluster メンバに対してローカルな存在です。どのような基本システムも、Virtual I/O Cache によるキャッシングをサポートすることができます。このキャッシング機能を使用するために、OpenVMS Cluster のライセンスは必要ではありません。



注意:

OpenVMS Cluster のメンバのいずれかでキャッシングが禁止されていると、その OpenVMS Cluster では、キャッシングは全く実行されません (キャッシングが許可されているノードでも同様です)。この状態は、キャッシングが禁止されている 1 つまたは複数のノードでキャッシングが許可されるか、またはそのようなノードがクラスタから削除されるまで続きます。

ロック・マネージャは、キャッシュを密接に管理します。1 つのノードが OpenVMS Cluster を離れると、キャッシュはフラッシュされます。複数のノードで書き込みアクセスを持ち、複数のノードで開かれているファイルは、キャッシュされません。

第5章 UETP によるシステムのテスト

この章では、UETP (ユーザ環境テスト・パッケージ) を使って、OpenVMS オペレーティング・システムが正しくインストールされているかどうかを調べる方法について説明します。

5.1 概要

ここでは、UETP が何を行うか、また UETP をどのように使用するかについて説明します。さらに、テストのためのシステムの設定、テストの実行、およびトラブルシューティングについての具体的な指示を行います。

この章の内容

この章では、次の作業について説明します。

作業項目	参照箇所
UETP の実行 (要約)	5.1.2 項 「UETP の使用方法 (概要)」
UETP を使用するための準備	5.2 項 「UETP を使用するための準備」
テストの対象となるデバイスの設定	5.3 項 「テストを行うデバイスの設定」
UETP の起動	5.5 項 「UETP の起動」
UETP の動作の停止	5.6 項 「UETP の動作の停止」
トラブルシューティング - 問題の識別と解決	5.8 項 「トラブルシューティング: 考えられる UETP エラー」

さらに、次の項目について説明します。

項目	参照箇所
UETP について	5.1.1 項 「UETP について」
トラブルシューティング (概要)	5.7 項 「トラブルシューティング: 概要」
UETP テストおよびフェーズ	5.9 項 「UETP テストおよびフェーズ」

5.1.1 UETP について

UETP は、OpenVMS オペレーティング・システムが正しくインストールされているかどうかをテストするソフトウェア・パッケージです。UETP は、日常の使用で発生し得る要求と同等の要求をシステムに対して行うことによって、通常のユーザ環境をシミュレートする一連のテストを行います。

UETP は診断プログラムではありません。つまり、すべての機能を徹底的にテストするわけではありません。UETP が回復不可能なエラーに遭遇せず、その実行を終えたということは、テストされたシステムが一応使用できるということを示しているだけです。

UETP は、すべての OpenVMS システムに共通なデバイスと機能を調べます。高水準言語コンパイラのようなオプションの機能についてはテストしません。次に、UETP がテストするシステム・コンポーネントを示します。

- ほとんどの周辺機器デバイス
- システムのマルチユーザ処理能力
- DECnet for OpenVMS ソフトウェア
- クラスタ単位のファイル・アクセスおよびファイル・ロック

5.1.2 UETP の使用方法 (概要)

この節では、UETP のすべてのフェーズを省略時の値を使用して実行する手順を要約して説明します。すでにこのテスト・パッケージを使用したことのある方は、この節を参照してください。より詳細な情報が必要な方は、5.2 項「UETP を使用するための準備」を参照してください。



注意:

OpenVMS Alpha システムまたは I64 システムで UETP を使用する場合、次の手順を行う前に SYS\$MANAGER 内の CREATE_SPECIAL_ACCOUNTS.COM コマンド・プロシージャを実行し、SYSTEST および SYSTEST_CLIG アカウントを作成しておいてください。CREATE_SPECIAL_ACCOUNTS.COM コマンド・プロシージャについての詳細は、『OpenVMS システム管理者マニュアル (上巻)』を参照してください。

1. 次のように、SYSTEST アカウントにログインする。

```
Username: SYSTEST
Password:
```



重要:

SYSTEST および SYSTEST_CLIG アカウントは特権を持っているので、権限のないユーザがこのアカウントを使用すると、システムの機密保護上、問題が発生することがあります。

2. ユーザ・プログラムが実行されていないこと、およびユーザ・ボリュームがマウントされていないことを確認する。



重要:

設計上、UETP はシステム資源を排他的に使用することを想定し、また、そのように要求します。この制約を無視すると、UETP は当該資源に依存するアプリケーションに影響を与えることがあります。

3. ログイン後、すべてのデバイスが次の状態であることをチェックする。
 - テストを行うすべてのデバイスは、電源が入っており、システムに対してオンラインであること。
 - スクラッチ・ディスクがマウントされており、初期化されていること。
 - ディスクの中に [SYSTEST] という名前のディレクトリが OWNER_UIC=[1,7] で存在すること (DCL の CREATE/DIRECTORY コマンドを使用して作成)。
 - テストを行うすべてのドライブにスクラッチ磁気テープ・リールが**物理的に**マウントされており、ラベル UETP で初期化されていること (DCL の INITIALIZE コマンドを使用)。磁気テープ・リールには、少なくとも 600 フィートのテープが入っていることを確認すること。
 - テストを行うすべてのドライブにスクラッチ・テープ・カートリッジが挿入されており、ラベル UETP で初期化されていること (DCL の INITIALIZE コマンドを使用)。
 - ライン・プリンタおよびハードコピー・ターミナルに十分な用紙が用意されていること。
 - ターミナル特性およびボー・レートが正しく設定されていること (各ターミナルのユーザ・ガイドを参照)。

通信デバイスの中には弊社のサポート担当者が設定しなければならないものもあるので注意する (5.3 項「テストを行うデバイスの設定」を参照)。

UETP のための準備中になんらかの問題が発生した場合は、次に進む前に 5.3 項「テストを行うデバイスの設定」を参照する。

4. UETP を起動するには、次のコマンドを入力し、Return を押す。

\$ @UETP

UETP は次の質問を行う。

Run "ALL" UETP phases or a "SUBSET" [ALL]?

Return を押して、大括弧で囲まれた省略時の応答を選択する。UETP は次の質問を行う。

How many passes of UETP do you wish to run [1]?

How many simulated user loads do you want [60]?

Do you want Long or Short report format [Long]?

各プロンプトに対して Return を押す。最後の質問に答えた後、UETP は一連のテストをすべて開始する。以降、完了するまで入力することはない。最後に、次のようなメッセージが表示される。

```
*****
*
*      END OF UETP PASS 1 AT 22-JUN-2004 16:30:09.38
*
*****
```



注意:

省略時の応答を使用せずに UETP を実行する場合には、5.5 項「UETP の起動」を参照してください。ここにはオプションが説明されています。

5. UETP の実行後、ログ・ファイルでエラーをチェックする。テストが成功終了した場合、OpenVMS オペレーティング・システムは適切に動作する状態である。
UETP が異常終了した場合は、トラブルシューティングの情報について 5.7 項「トラブルシューティング：概要」を参照する。



注意:

UETP の実行後、Error Log Viewer (ELV) または System Event Analyzer (SEA) を使用して、UETP 実行中に発生したハードウェア障害をチェックしてください。ELV の実行についての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。SEA についての詳細は、Web サイト h18023.www1.hp.com/support/svctools/webes/sea_uj.pdf を参照してください。

5.2 UETP を使用するための準備

この節では、UETP を実行するための詳細な情報について説明します。

- ログイン
- [SYSTEST] ディレクトリの使用方法

5.2.1 ログイン

システム管理者から SYSTEST のパスワードを教えてください。次のように入力し、コンソール・ターミナルから SYSTEST アカウントにログインします。

Username: **SYSTEST**
Password:



注意:

SYSTEST には特権があるので、権限のないユーザがこのアカウントを使用すると、システムの機密保護上、問題が発生することがあります。

SYSTEST アカウントからテストを実行しないと、UETP は失敗します。また、コンソール・ターミナル以外のターミナルから UETP を実行しようとする時、現在使用しているターミナルはテストには使用できないというエラー・メッセージがデバイス・テスト・フェーズで表示されます。このメッセージは無視しても構いません。

SYSTEST アカウントにログインした後、コマンド SHOW USERS を入力して、ユーザ・プログラムが実行されていないことと、ユーザ・ボリュームがマウントされていないことを確認してください。UETP は、システム資源を排他的に使用することを想定しています。この制約を無視すると、UETP はその資源に依存するアプリケーションに影響を与えることがあります。



注意:

UETP を実行するとき発生し得る、クォータ、特権、またはアカウントの間違いなどの問題を識別および解決するための情報については、5.8.2 項「クォータ、特権、アカウントの間違い」を参照してください。UETP を実行する前に、この節に目を通しておいてください。

5.2.2 SYSTEST ディレクトリの使用方法

ログインに成功すると、省略時のディレクトリはシステム・ディスク上の [SYSTEST] となります。UETP は、UETP コマンド・プロシージャ (UETP.COM) が使用するファイル、および UETP がテスト中に使用する一時ファイルをこのディレクトリに格納します。

典型的なシステムでは、論理名 SYS\$TEST は次のように定義されています。

```
$ SHOW LOGICAL SYS$TEST
  "SYS$TEST" = "SYS$SYSROOT:[SYSTEST]" (LNM$SYSTEM_TABLE)
```

スクラッチ・ディスクなどの特定のディスクを UETP でテストするには、そのディスクに [SYSTEST] ディレクトリまたは [SYS0.SYSTEST] ディレクトリを作成します。スクラッチ・ディスクのテストのための設定については、5.3.3 項「UETP のディスク上での動作」を参照してください。

5.3 テストを行うデバイスの設定

この節の説明に従って、ログイン後にシステム上のデバイスを UETP テスト用に設定します。なお、この節で述べられているすべてのデバイスがユーザのシステムにも接続されているとは限らないので注意してください。

5.3.1 デバイスのチェック

UETP が使用するすべてのデバイスが次の状態であるかチェックしてください。

- テストを行うすべてのデバイスは、電源が入っており、システムに対してオンラインであること。
- スクラッチ・ディスクが初期化されており、マウントされていること。
- ディスクの中に [SYSTEST] という名前のディレクトリが OWNER_UIC=[1,7] で存在すること。[SYSTEST] ディレクトリがディスク上に存在しない場合は、DCL の CREATE/DIRECTORY コマンドで作成。
- テストを行うすべてのドライブにスクラッチ磁気テープ・リールが物理的にマウントされており、ラベル UETP で初期化されていること (DCL の INITIALIZE コマンドを使用)。磁気テープ・リールには、少なくとも 600 フィートのテープが入っていることを確認すること。
- テストを行うすべてのドライブにスクラッチ・テープ・カートリッジが挿入されており、ラベル UETP でマウントおよび初期化されていること (DCL の INITIALIZE コマンドを使用)。

- ライン・プリンタおよびハードコピー・ターミナルに十分な用紙が用意されていること。
- ターミナル特性およびボー・レートが正しく設定されていること (各ターミナルのユーザ・ガイドを参照)。

この節で述べる通信デバイスの中には弊社のサポート担当者が設定しなければならないものもあるので注意してください。

5.3.2 必要なシステム・ディスク領域

UETP を実行する前に、システム・ディスクの少なくとも 1200 ブロックが使用できることを確認してください。ロード・テスト・プロセスを 20 より多く実行するシステムでは、少なくとも 2000 ブロックが必要です。UETP の複数パスを実行する場合、ログ・ファイルは省略時のディレクトリに蓄積され、後続のパスで使用できるディスク領域が減少します。

ディスク・クォータがシステム・ディスクに対して使用可能に設定されている場合は、UETP を実行する前に、ディスク・クォータを使用不能に設定しておいてください。

5.3.3 UETP のディスク上での動作

UETP のディスク・テスト・フェーズでは、テストを行うすべてのディスク上で使用できるほとんどの空き領域は、次のような方法で使用されます。

- テストを行うすべてのディスク上において、デバイス・テスト・フェーズは2つのファイルを作成しようとする。この2つのファイルのサイズは、そのディスク上で使用できる空き領域の大きさによって異なる。各ファイルの作成には通常、ディスク上の空き領域の0.1%を使用する。しかし、ディスクがほとんどいっぱいの場合、このテストは5ブロックのファイルを作成する。5ブロックのファイルが作成できなければ、テストは失敗する。ディスク領域の不足のためにテストが失敗するのは、最初のファイルの作成時だけである。
- テストはランダムにデータのブロックをファイルから読み込んだり、ファイルに書き込んだりする。各ファイルへ20回書き込むごとに、テストはそのファイルを拡張しようとする。このファイルの拡張のサイズは、空きディスク領域の5%である。なお、ファイルが5ブロックで作成されていた場合は、このサイズも5ブロックになる。この拡張プロセスは、全ファイルの領域が合わせて空きディスク領域の75%に達するまで継続される。

このように、断片化されたファイルを作成および拡張することによって、UETP はディスクを調査します。この調査によって、クォータを超過しているか、またはディスクがいっぱいになっているかをチェックでき、使用できるディスク領域の大きさも調整できます。

他のディスクと同様に、シャドウ・セットやボリューム・セットもUETPでテストできますが、UETINIDEV (UETP の初期化) の間に、個々のメンバがテスト不能としてリストされることが予想されます。UETINIDEV は、システム・ディスク (UETDISK00) のパス中にシャドウ・セットでテストすると、エラーを表示しますが、シャドウ・セットはテスト可能とリストされます。ボリューム・セットでテストすると、相対ボリューム番号1以外ではエラーが出て、UETINIDEV の最後にテスト不能とリストされます。

5.3.4 ディスク・ドライブの準備

次の手順に従って、システム上の各ディスク・ドライブをUETPテスト用に準備してください。

1. スクラッチ・ディスクをドライブに入れ、ドライブを回転させる。スクラッチ・ディスクが使用できない場合は、空き領域が十分にあるディスクで代用する。どのボリューム上においても、UETP は既存のファイルを上書きしない。スクラッチ・ディスクの中に保管しておきたいファイルがある場合は、ディスクを初期化しないで、ステップ3に進む。
2. 保存したいファイルがディスクに存在しない場合には、そのディスクを初期化する。次に例を示す。

```
§ INITIALIZE DUA1: TEST1
```

このコマンドは、DUA1 を初期化し、TEST1 というボリューム・ラベルをそのディスクに割り当てる。同じラベルのボリュームが存在してはならない。

3. ディスクをマウントする。次に例を示す。

```
§ MOUNT/SYSTEM DUA1: TEST1
```

このコマンドは、TEST1 というラベルのボリュームを DUA1 上にマウントする。/SYSTEM 修飾子は、システム上のすべてのユーザが使用できるボリュームを作成していることを示している。

4. ディスクのテスト時、UETP は [SYSTEST] ディレクトリを使用する。ディレクトリ [SYSTEST] がボリュームに存在しない場合、このディレクトリを作成しなければならない。次に例を示す。

```
§ CREATE/DIRECTORY/OWNER_UIC=[1,7] DUA1:[SYSTEST]
```

このコマンドは [SYSTEST] ディレクトリを DUA1 上に作成し、利用者識別コード (UIC) として [1,7] を割り当てる。UETP を実行するためには、ディレクトリの UIC は [1,7] でなければならない。

マウントしたディスクにルート・ディレクトリ構造が存在する場合には、[SYS0.] ツリーの中に [SYSTEST] ディレクトリを作成することができます。

5.3.5 磁気テープ・ドライブ

次の手順に従って、テストを行う磁気テープ・ドライブを設定します。

1. 少なくとも 600 フィートの磁気テープを持つスクラッチ磁気テープをテープ・ドライブに入れる。書き込み可能リングが装着されていることを確認する。
2. 磁気テープの位置を BOT (テープの開始) に合わせ、そのドライブをオンラインにする。
3. すべてのスクラッチ磁気テープをラベル UETP で初期化する。たとえば、スクラッチ磁気テープを MUA1 上に物理的にマウントしている場合、次のコマンドを入力し、Return を押す。

```
§ INITIALIZE MUA1: UETP
```

テストを行うテープには UETP というラベルが付いていなければならない。安全のため、UETP は MOUNT コマンドでマウントしたテープはテストしない。

磁気テープの初期化中に問題が発生した場合、または磁気テープへのアクセスに問題がある場合は、『OpenVMS DCL ディクショナリ』の DCL の INITIALIZE コマンドの説明を参照してください。

5.3.6 テープ・カートリッジ・ドライブ

次の手順に従って、テストを行うテープ・カートリッジ・ドライブを設定します。

1. スクラッチ・テープ・カートリッジをテープ・カートリッジ・ドライブに入れる。
2. テープ・カートリッジを初期化する。次に例を示す。

```
§ INITIALIZE MUA0: UETP
```

テストを行うテープ・カートリッジには UETP というラベルが付いていなければならない。安全のため、UETP は MOUNT コマンドでマウントしたテープ・カートリッジはテストしない。

テープ・カートリッジの初期化中に問題が発生した場合、またはテープ・カートリッジへのアクセスに問題がある場合は、『OpenVMS DCL ディクショナリ』の DCL の INITIALIZE コマンドの説明を参照してください。

TLZ04 テープ・ドライブ

初期化フェーズ中、TLZ04 ユニットが UETTAPE00 テストを完了するまでの時間は 6 分に設定されます。この時間内に UETTAPE00 テストが完了しなければ、UETP は次のようなメッセージを表示します。

```
-UETP-E-TEXT, UETTAPE00.EXE testing controller MKA was stopped ($DELPRC)
      at 16:23:23.07 because the time out period (UETP$INIT_TIMEOUT)
      expired or because it seemed hung or because UETINIT01 was aborted.
```

このタイムアウト値を増やすには、UETP を実行する前に、次のようなコマンドを入力します。

```
$ DEFINE/GROUP UETP$INIT_TIMEOUT "0000 00:08:00.00"
```

この例では、初期化タイムアウト値を 8 分に定義しています。

5.3.7 コンパクト・ディスク・ドライブ

UETP をコンパクト・ディスク・ドライブ上で実行する場合には、まず、コンパクト・ディスク・ドライブ・ユニットに添付されているテスト・ディスクをロードしなければなりません。

5.3.8 光ディスク・ドライブ

UETP を RV60 ドライブ上で実行するには、次の手順に従って、RV64 光ディスク記憶システムを設定します。

1. Jukebox Control Software (JCS) を使って、光ディスクをすべての RV60 ドライブにロードする。JCS は、RV64 に添付されている OpenVMS オペレーティング・システム上のレイヤード製品で、ディスクをロードおよびアンロードするロボット・アームを制御するものである。
2. 光ディスクをラベル UETP で初期化する。マウントは行わない。

UETP は、RV64 に存在するすべての RV60 を同時にテストします。テープ・テストと異なり、UETP はテスト終了時に光ディスクを再初期化しません。

5.3.9 ターミナルおよびライン・プリンタ

UETP でテストを行うターミナルおよびライン・プリンタは、電源が入っており、システムに対してオンラインでなければなりません。ライン・プリンタおよびハードコピー・ターミナルに、十分な用紙が用意されていることを確認します。用紙の量は、実行する UETP パスの数によって異なります。パスごとに 2 枚の用紙がライン・プリンタおよびハードコピー・ターミナルに必要です。

すべてのターミナルについて、ポー・レートが正しく設定されており、特性が適切に割り当てられていることをチェックします (各ターミナルのユーザ・ガイドを参照してください)。

デバイスをスプールしたり、キューに割り当てると、UETP の初期化フェーズで失敗し、テストが行われません。

5.3.10 DR11-W データ・インタフェース (VAX のみ)

DR11-W データ・インタフェースは、内部論理ループバック・モードを使って、モジュール・コネクタ、ケーブル、トランシーバを除くすべての機能をテストします。



重要:

DR11-W データ・インタフェースを UETP テスト用に設定できるのは弊社のサポート担当者だけです。

この動作中にはランダムな外部パターンが作成されるので、テストを行うユーザ・デバイスまたは他のプロセッサは、場合によっては、テストが完了するまで DR11-W データ・インタフェースから分離しておく必要があります。

DR11-W データ・インタフェースを適切にテストするには、E105 スイッチパックを次のように設定しておかなければなりません。

スイッチ 1	スイッチ 2	スイッチ 3	スイッチ 4	スイッチ 5
オフ	オン	オフ	オフ	オン

UETP のテストが完了したら、DR11-W データ・インタフェースを適切な動作構成に戻します。

5.3.11 DRV11-WA データ・インタフェース (VAX のみ)

DRV11-WA データ・インタフェースは、汎用の 16 ビット・パラレルのダイレクト・メモリ・アクセス (DMA) データ・インタフェースです。



重要:

DRV11-WA データ・インタフェースを UETP テスト用に設定できるのは弊社のサポート担当者だけです。

MicroVAX コンピュータ上の DRV11-WA ドライバを UETP テスト用に準備するためには、次の状態を確認してください。

- DRV11-WA ボード上のジャンパが W2, W3, および W6 に設定されていること。
- ループバック・ケーブルが DRV11-WA ボードに接続されていること。
- DRV11-WA ボードがスロット 8 から 12 まで占有していること。DRV11-WA が他の場所にある場合、タイムアウト・エラーが発生することがある。

UETP テストが完了したら、DRV11-WA を適切な動作構成に戻します。

5.3.12 DR750 または DR780 (DR32 インタフェース) (VAX のみ)

DR32 (DR750 または DR780) デバイスは、VAX プロセッサの内部メモリ・バスを、DR32 デバイス間接続 (DDI) と呼ばれるユーザ・アクセス可能バスに接続するインタフェース・アダプタです。



重要:

DR750 または DR780 を UETP テスト用に設定できるのは弊社のサポート担当者だけです。

次の手順に従って、DR750 または DR780 を UETP テスト用に準備してください。

1. DR780 マイクロコード・ファイル XF780.ULD を診断媒体から SYS\$SYSTEM にコピーする。DR780 マイクロコード・キットに添付されたドキュメントに記載されている手順に従うこと。
2. DR780 の電源を切る。
3. 次の DR780 背面ジャンパを変更する。
 - a. W7 および W8 からジャンパを外す。
 - b. E04M1 から E04R1 までにジャンパを追加する。
 - c. E04M2 から E04R2 までにジャンパを追加する。
4. DDI ケーブルを DR780 から外す。このケーブルは、BC06V-nn ケーブルの場合と BC06R-nn ケーブルの場合がある。前者はそのまま外すことができるが、後者を外すときは、DR780 の背面からバドル・カードを取り外さなければならない。
5. もう一度、DR780 の電源を入れる。

UETP テストが完了したら、DR750 または DR780 を適切な動作構成に戻します。

5.3.13 2 台目の LPA11-K デバイス

LPA11-K デバイスが 2 台存在する場合は、各デバイスのシステムワイドの論理名が SYS\$MANAGER:LPA11\$STRT.COM ファイルに指定されていることを確認してください。最初の LPA11-K デバイスに対する論理名は LPA11\$0 で、2 番目の LPA11-K デバイスに対する論理名は LPA11\$1 でなければなりません。

5.3.14 テストを行わないデバイス

UETP は、次のデバイスに関してはテストを行いません。これらのデバイスの状態は UETP の実行に全く影響を与えません。

- オペレータとの対話が必要なデバイス (カード・リーダなど)
- ソフトウェア・デバイス (ヌル・デバイスおよびローカル・メモリ・メールボックスなど)

UETP は、UDA、HSC、または CI デバイスについてはテストを行いません。これらのデバイスは、ディスク、磁気テープ、および DECnet for OpenVMS のテストで暗黙にテストされます。

また、UETP はコンソール・ターミナルまたはコンソール・ドライブについてもテストを行いません。システムをブートし、ログインして、UETP を起動すれば、これらのデバイスが使用できるかどうかを確認できます。

5.3.15 OpenVMS Cluster のテスト

OpenVMS Cluster 環境で UETP を実行する前には、SYSTEST_CLIG アカウントをチェックしてください。SYSTEST_CLIG アカウントは、SYSTEST アカウントに似ていますが、クラスタ統合テストのみ行います。次に、SYSTEST_CLIG アカウントの要件を示します。

- このアカウントは、OpenVMS クラスタ中の各システム上の利用者登録ファイルに、作成されたときのままの形で存在しなければならない (5.1.2 項「UETP の使用方法 (概要)」の注意を参照)。



注意:

SYSTEST_CLIG アカウントは、作成しただけでは使用不可になっています。UETP を実行する前に、SYSTEMST_CLIG アカウントを使用可能状態にし、ヌル・パスワードを割り当てなければなりません。

SYSTEST_CLIG アカウントを再び使用可能にするには、次のコマンドを入力する。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> MODIFY /FLAGS=NODISUSER /NOPASSWORD SYSTEST_CLIG
UAF> EXIT
```



注意:

テスト終了後、SYSTEST_CLIG アカウントは使用不可にしておいてください。

SYSTEST_CLIG アカウントを使用不可にするには、次のようなコマンドを入力する。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> MODIFY /FLAGS=DISUSER SYSTEST_CLIG
UAF> EXIT
```

- SYSTEST_CLIG アカウントの特権およびクォータは、SYSTEST アカウントと同じでなければならない。

クラスタ統合テスト・フェーズの場合、通常の UETP テスト・フェーズの要件に加えて、さらに準備することがあります。次に、クラスタ統合テストのために必要な追加の要件を示します。

1. ユーザのシステムがクラスタのメンバであること。メンバでない場合、UETP はメッセージを表示し、テストを実行しない。
2. ユーザのシステムが、クラスタ中の他のシステムと同じデッドロック検出インターバルを使用していること (デッドロック検出インターバルは、SYSGEN パラメータの DEADLOCK_WAIT で設定する。通常は、省略時の設定 (10 秒) から変更されていない)。
3. テストを行うすべてのシステムの SYS\$TEST 中に、ファイル UETCLIG00.COM および UETCLIG00.EXE が存在すること。
4. DECnet for OpenVMS が クラスタ・ノード間で設定されていること。UETP は、DECnet for OpenVMS を使って、上記ノード上にプロセスを作成する。テストで行われるチェックは、SYSTEST_CLIG プロセスを作成する能力、および、DECnet for OpenVMS ソフトウェアを使って上記プロセスと通信する能力によって異なる。
5. ノード名が DECnet データベースに定義されていることを確認する。
6. すべてのオペレータ・ターミナル (OPAO:) は、ブロードキャスト・メッセージを受信できなければならない。BROADCAST 特性を設定するには、次のコマンドを入力する。

```
$ SET TERM/BROADCAST/PERM OPAO:
```

オペレータ・ターミナル (OPAO) に NO BROADCAST ターミナル特性が設定されているノードでは、クラスタ・テスト中に、次のメッセージが表示される。

```
*****
* UETCLIG00master      *
* Error count = 1      *
*****
-UETP-E-TEXT, 0 operator consoles timed out on the cluster test warning
and 1 operator console rejected it.
-UETP-E-TEXT, Status returned was,
"%SYSTEM-F-DEVOFFLINE, device is not in configuration or not
available"
```

7. クラスタ内の各ノード (OpenVMS および HSC) において、[SYSTEST] または [SYS0.SYSTEST] ディレクトリが、クラスタで使用できるディスク上に存在しなければならない。テストは、UETP ディスク・テストと同じディレクトリを使って、各クラスタ・ノード上にファイルを作成し、そのクラスタ中の他の OpenVMS ノードがそのファイルに共有アクセスできるかどうかを調べる。このようなディレクトリは、ノードごとに1つずつ必要である。テストは、1つのファイルが処理されるたびに、次のクラスタ・ノードに続く。
8. 省略時の設定で、UETP クラスタ・フェーズは、デッドロック・テスト、ディスク・テスト、およびファイル・アクセス・テスト用に、実行中のクラスタから3つのノードを選択する。しかし、すべてのクラスタ・ノードをテストしたい場合は、UETP を起動する前に、次のコマンドを入力する。

```
$ DEFINE/GROUP UETP$CTMODE ALL
```

5.3.16 小規模ディスク・システムのテスト方法

小規模なシステム・ディスク (RZ23L など) に OpenVMS VAX オペレーティング・システムをインストールすると、UETP を実行するのに必要な 1200 ブロックの空きディスク領域がありません。システム・ディスクに 1200 ブロックの空き領域がない場合は、UETP を実行する前に、VMSTAILOR を使って、いくつかのファイルをシステム・ディスクから削除します。VMSTAILOR の使用方法についての指示は、使用しているシステムの OpenVMS のアップグレードおよびインストール・マニュアルを参照してください。

5.3.17 DECnet for OpenVMS フェーズ

UETP の DECnet for OpenVMS フェーズは、他のテストより多くのシステム資源を使用します。しかし、最も負荷の低いノード上でテストを行うことで、他のユーザへの影響を最小限に抑えることができます。

省略時の設定で、ファイル UETDNET00.COM は、DECnet テストを行うノードを指定します。異なるノードで DECnet テストを行う場合は、UETP を実行する前に、次のコマンドを入力します。

```
$ DEFINE/GROUP UETP$NODE_ADDRESS node_address
```

このコマンドは、グループ論理名 UETP\$NODE_ADDRESS に、UETP の DECnet フェーズを実行したいユーザの領域内にあるノードのノード・アドレスを割り当てます。

次に例を示します。

```
$ DEFINE/GROUP UETP$NODE_ADDRESS 9.999
```

UETP を実行する前に次のコマンドを入力すると、異なるノード上で DECnet for OpenVMS テストを行うことができます。

```
$ DEFINE/GROUP UETP$NODE_NAME "node" "username password"
```



注意:

論理名 UETP\$NODE_ADDRESS を使用すると、UETP は NCP (ネットワーク制御プログラム) で最初に見つかったアクティブなサーキットだけをテストします。使用しなかった場合、UETP はアクティブでテスト可能なサーキットをすべてテストします。

UETP の実行時、ルータ・ノードは、UETP\$NODE_ADDRESS または UETP\$NODE_NAME で定義されたノードとユーザのノードとの間で接続を確立しようとしています。ユーザのノードとルータ・ノード間の接続がビジー状態であったり、存在しないことがあります。このような場合には、システムは次のようなエラー・メッセージを表示します。

```
%NCP-F-CONNEX, Unable to connect to listener  
-SYSTEM-F-REMRSRC, resources at the remote node were insufficient
```

```
%NCP-F-CONNEX, Unable to connect to listener  
-SYSTEM-F-NOSUCHNODE, remote node is unknown
```

5.3.18 DECnet Phase 5 の論理名

DECnet Phase 5 システムでは、UETP\$NODE_NAME 論理名を定義し、ログイン情報に含めなければなりません。ノードを番号で指定することはできません (ピリオド (.) が紛らわしいためです)。

UETP\$NODE_NAME 論理名を定義するには、次のコマンドを使用します。

```
$ DEFINE/SYSTEM UETP$NODE_NAME "gamev5" "systest" "" <password>  
$ @UETP
```

```
Welcome to OpenVMS UETP Version X9Y4-SSB  
  
%UETP-I-ABORTC, UETINIT00 to abort this test, type ^C  
  
You are running on a AlphaServer 2100 5/250 CPU.  
The system was booted from _$21$DKA100:[SYS1.].  
  
Run "ALL" UETP phases or a "SUBSET" [ALL]? S  
  
You can choose one or more of the following phases:  
  
DEVICE, LOAD, DECNET, CLUSTER  
  
Phase(s): dec  
  
How many passes of UETP do you wish to run [1]?  
Do you want Long or Short report format [Long]?  
  
UETP starting at 5-SEP-2003 14:10:17.71 with parameters:  
DECNET phases, 1 pass, 10 loads, long report.  
  
%UETP-I-BEGIN, UETDNET00 beginning at 5-SEP-2003 14:10:17.86  
%UETP-I-BEGIN, UETDNET00_00000 beginning at 5-SEP-2003 14:10:17.94
```

```

**** UETDNET00 BEGINNING AT 5-SEP-2003 14:10:18.22 ****
%UETP-I-TEXT, Testing remote node gamev5
%UETP-I-BEGIN, Remote circuit testing beginning at 5-SEP-2003 14:10:20.21
%UETP-I-BEGIN, UETDNET01 beginning at 5-SEP-2003 14:10:20.31
%UETP-I-BEGIN, GAMEV5TST_00000 beginning at 5-SEP-2003 14:10:20.51
%UETP-I-BEGIN, GAMEV5TST_00001 beginning at 5-SEP-2003 14:10:20.66
%UETP-W-TEXT, The process -GAMEV5TST_00000- returned a final status of:
%DELETE-W-SEARCHFAIL, error searching for !AS
%UETP-I-ENDED, GAMEV5TST_00000 ended at 5-SEP-2003 14:10:30.06
%UETP-W-TEXT, The process -GAMEV5TST_00001- returned a final status of:
%DELETE-W-SEARCHFAIL, error searching for !AS
%UETP-I-ENDED, GAMEV5TST_00001 ended at 5-SEP-2003 14:10:30.07
%UETP-I-ENDED, UETDNET01 ended at 5-SEP-2003 14:10:30.24
%UETP-I-ENDED, Remote circuit testing ended at 5-SEP-2003 14:10:30.28
%UETP-I-ENDED, UETDNET00_00000 ended at 5-SEP-2003 14:10:31.13
%UETP-I-ENDED, UETDNET00 ended at 5-SEP-2003 14:10:31.19
*****
*
*   END OF UETP PASS 1 AT 5-SEP-2003 14:10:31.59
*
*****

```

5.3.19 ベクタ・プロセッサおよび VVIEF (VAX のみ)

UETP は、ロード・フェーズ中に、インストールされて使用可能なベクタ・プロセッサを自動的にロードし、デバイス・テスト・フェーズ中に、インストールされた使用可能なベクタ・プロセッサを自動的にテストします。

ベクタ・プロセッサがシステムで使用可能な場合、次のようなコマンドを入力して、VP 番号をチェックしてください。

```

$ x = F$GETSYI ("VP_NUMBER")
$ SHOW SYMBOL x

```

x の値に 3 をかけます。その結果がアカウントの PRCLM 値より大きい場合、戻された結果に一致するように、SYSTEST アカウントの PRCLM クォータを増やさなければなりません。詳細は第13章「特殊処理環境の管理」を参照してください。

しかし、UETP は、ロード・フェーズ中に、VAX ベクタ命令エミュレーション機能 (VVIEF) をロードすることができないので、VVIEF を自動的にテストできません。VVIEF をテストするには、UETP を実行する前に、次の手順を行わなければなりません。

1. ファイル UETCONT00.DAT を編集し、次の行を追加する。

```

Y   Y   UETVECTOR.EXE  "DEVICE_TEST"

```

2. システムをブートしたときに VVIEF が起動されたかどうかを確認する。VVIEF が起動されたかどうかを確認するには、次の DCL コマンドを入力する。

```

$ X = F$GETSYI ("VECTOR_EMULATOR")
$ SHOW SYMBOL X

```

システムが 1 という値を表示した場合、VVIEF はロードされている。システムが 0 という値を表示した場合、VVIEF はロードされていない。

RUN コマンドを使って、VVIEF テストを個々のテストとして実行することができます (5.9.2 項「デバイス・テスト・フェーズ」を参照)。

5.4 バッチでの UETP の実行

バッチで UETP を実行すると便利な場合があります。バッチで実行する方法を、次の例で示します。

```

$ submit SYS$COMMON:[SYSTEST]uetp/param: ("load", "1000", "20", "long") -
_ $ /queue:whamoo_batch/username:systest/log:whamoo.log

```

この例では、パスを 1000、負荷を 20 と指定しています。

5.5 UETP の起動

ログイン後、システムとデバイスの準備が終わったら、テストを開始することができます。UETP を起動するには、次のコマンドを入力し、Return を押します。

```
$ @UETP
```

UETP は次のプロンプトを表示します。

```
Run "ALL" UETP phases or a "SUBSET" [ALL]?
```

スタートアップ・ダイアログにおいて、大括弧の中の値は省略時の値、つまり、Return を押したときに選択できる値を示します。

初めて UETP を実行する場合は、省略時の値 (ALL) を選択して、すべてのフェーズを実行することをお勧めします。ALL を選択すると、UETP はさらに 3 つの質問を表示します。この質問については、5.5.2 項「1 つのフェーズの実行と複数のフェーズの実行」から 5.5.4 項「レポート形式」までを参照してください。すべてのテスト・フェーズを実行する場合は、次の項は関係ありません。

5.5.1 フェーズのサブセットの実行方法

フェーズを 1 つだけ実行するには、次のプロンプトに SUBSET または S を入力します。

```
Run "ALL" UETP phases or a "SUBSET" [ALL]?
```

SUBMIT または S を入力した場合、UETP は、次のように、実行したいフェーズの入力を求めてきます。

You can choose one or more of the following phases:

```
DEVICE, LOAD, DECNET, CLUSTER
```

Phases (s) :

省略時の値はないので、上記リストの中から 1 つまたは複数の名前を入力してください。複数のフェーズを入力する場合には、それぞれをスペースまたはコンマで区切ります。

LOAD フェーズが選択の中にある場合、UETP は 3 つのプロンプトを表示します。

```
How many passes of UETP do you wish to run [1]?
```

```
How many simulated user loads do you want [n]?
```

```
Do you want Long or Short report format [Long]?
```

LOAD フェーズを選択しなかった場合、1 番目と 3 番目のプロンプトの 2 つしか表示されません。

次の 3 つの項は、これらの質問にどう答えるかを説明しています。質問に答えた後、選択したフェーズの実行が始まります。

5.5.2 1 つのフェーズの実行と複数のフェーズの実行

最後のプロンプトに省略時の ALL またはフェーズのサブセットを指定した場合、UETP は次のようなメッセージを表示します。

```
How many passes of UETP do you wish to run [1]?
```

テストは、何度でも繰り返して実行することができます。プロンプトに 1 を入力すれば (または省略時の設定で Return を押せば)、UETP はテストを 1 回実行して終了します。1 より大きな数を指定すれば、UETP は指定した回数だけテストを繰り返します。

システムが動作しているかどうかをチェックする場合は、UETP を 1 回だけ実行します。連続使用におけるシステムのレスポンスを評価する場合は、UETP を複数回実行します。たとえば、サービス技術者などが新しくインストールしたシステムが動作するかを確認するのであれば、

1 回か 2 回だけ UETP を実行すれば十分です。また、製造技術者などはシステム統合およびテストの一部としてシステムを何時間も実行することがあります。

UETP を複数回実行するように指定した場合は、コンソール・ログを短く表示させることもできます (5.5.4 項「レポート形式」を参照)。1 回の実行ごとに 2 ページずつ出力されるので、ライン・プリンタおよびハードコピー・ターミナルには十分な用紙を用意しておいてください。

5.5.3 ロード・テスト用のユーザ負荷の定義

フェーズの回数を指定した後、UETP は次のプロンプトを表示します。

```
How many simulated user loads do you want [n]?
```



注意:

UETP がこのプロンプトを表示するのは、LOAD フェーズを選択したときだけです。たとえば、(すべてのフェーズを実行して) 暗黙に選択した場合も、(フェーズのサブセットを実行するときに LOAD フェーズを) 明示的に指定した場合も含まれます。

ロード・テストは、複数のユーザ(独立プロセス)がシステム資源をめぐる競争する状況をシミュレートします。このプロンプトに対して、このテストでシミュレートするユーザ数を入力します。大括弧の中の数は、UETP がユーザのシステムから算出した省略時の値です。したがって、省略時の値は、ユーザのシステムが割り当てたメモリの量、ページング領域、およびスワップ領域によって異なります。

省略時の値が最も適切な選択ですが、このプロンプトにユーザが値を指定することにより、省略時の値を増やしたり減らしたりすることができます。しかし、あまり数を増やしすぎると、資源が不十分になり、テストが失敗することがあるので注意してください。

UETP 実行時にユーザ負荷を求める公式を表示する方法については、5.7.2 項「UETP 出力の中断」を参照してください。

5.5.4 レポート形式

次のプロンプトでは、長いレポート形式を使用するか短いレポート形式を使用するかを選択できます。

```
Do you want Long or Short report format [Long]?
```

5.5.4.1 長いレポート形式

長いレポート形式(省略時の設定)を選択した場合、UETP はコンソール・ターミナルに次の情報を送信します。

- すべてのエラー・メッセージ
- すべてのフェーズおよびテストの開始時に作成された出力
- すべてのフェーズおよびテストの終了時に作成された出力

上記質問への応答にかかわらず、UETP はすべての出力を UETP.LOG ファイルに記録します。

ほとんどの場合、大量の出力をターミナルに書き込むというのは有効であるとは言えません。たとえば、UETP をハードコピー・ターミナルから実行する場合、出力のプリントに時間がかかってしまい、テスト自身の進行が遅くなる可能性があります。実行が 1 回だけなら、この遅延も気にならないかもしれませんが、ハードコピー・ターミナルから UETP を複数パス実行する場合は、短いレポート形式の方がいいでしょう。

5.5.4.2 短いレポート形式

短いレポート形式を要求すると、UETP は、エラー・メッセージやフェーズの開始と終了時の通知などの状態情報だけをコンソールに表示します。この情報は、UETP が正常に処理しているかどうかを判断する材料となります。短い形式のコンソール・ログがなんらかの問題を示し

た場合は、ファイル UETP.LOG を見れば、より詳細な情報を入手できます。UETP.LOG には、コンソールに表示された状態情報に加えて、さまざまなフェーズで作成されたすべての出力も入っています。

レポート形式の選択後、UETP は一連のテストを開始し、実行を始めます。UETP が異常終了した場合は、トラブルシューティングの情報について、5.7 項「トラブルシューティング：概要」を参照してください。

5.6 UETP の動作の停止

UETP パスの最後に、マスタ・コマンド・プロシージャ UETP.COM はパスの終了時刻を表示します。さらに、UETP.COM は再起動すべき UETP を決定します。テスト・パッケージを起動するとき、複数のパスを要求することも可能です (5.5.2 項「1 つのフェーズの実行と複数のフェーズの実行」を参照)。

UETP の実行が完全に終了すると、UETP.COM は一時的ファイルの削除などのクリーンアップ作業を行います。

Ctrl/Y または Ctrl/C を押すと、UETP が正常終了する前に、UETP の実行を中断することができます。ただし、UETP の実行が正常終了した場合は、UETP がテストのために作成したさまざまなファイルの削除も行われます。Ctrl/Y または Ctrl/C を押して UETP の実行を中断した場合、これらのクリーンアップ手続きが中断されたり、全く行われなかったりする可能性もあります。

このような制御文字の影響は、実行している UETP の部分によって異なります。UETP の編成およびその構成要素については、5.9 項「UETP テストおよびフェーズ」を参照してください。

5.6.1 Ctrl/Y の使用方法

Ctrl/Y を押すと、UETP の実行を強制終了します。ただし、[SYSTEST] 中のファイルおよびネットワーク・プロセスのクリーンアップは完了しないので注意してください。

個々のテスト・イメージを実行している場合に Ctrl/Y を押すと、現在の UETP テストを中断し、一時的に制御をコマンド・インタプリタに戻します。テストが中断されている間は、コマンド・インタプリタ内で実行され、現在のイメージを終了させない DCL コマンドのサブセットを入力することができます。

5.6.2 DCL コマンドの使用方法

『OpenVMS ユーザーズ・マニュアル』に、コマンド・インタプリタ内で使用できるコマンドの表が掲載されています。さらに、次のコマンドも入力できます。

- CONTINUE コマンドは、中断したところからテストを継続する (クラスタ・テストの実行中は除く)。
- STOP コマンドは、テストを終了する。テストは強制終了し、制御はコマンド・インタプリタに戻る。
- EXIT コマンドは、クリーンアップ処理を行ってから、テストを終了する (クラスタ・テストの実行中は除く)。制御はコマンド・インタプリタに戻る。

コマンド・インタプリタ内で実行されない DCL コマンドを入力すると、クリーンアップ処理が行われ、テストが完全に終了してから、その DCL コマンドが実行されます。

5.6.3 Ctrl/C の使用方法

Ctrl/C を押すと、UETP の実行を中断されます。Ctrl/C を押した後は、同じテスト・フェーズを継続することはできません。UETP は自動的に、マスタ・コマンド・プロシージャの次のフェーズに移ります。

UETP フェーズの中には、Ctrl/C を押すと、すべてのアクティビティをクリーンアップし、即座に終了するものもあります。これらのフェーズは、開始時に、次のようなメッセージが表示されます。

```
%UETP-I-ABORTC, 'testname' to abort this test, type ^C
```

上記メッセージを表示しないフェーズは、そのフェーズ内で起動されたすべてのプロセスを終了します。これらのプロセスでは、通常のクリーンアップ処理が行われなかったことがあります。しかし、個々のテスト・イメージを実行している場合、Ctrl/Cを使ってそのイメージの実行を終了し、クリーンアップ処理を完了させることができます。

クラスタ・テストの場合、Ctrl/C はクリーンアップ処理を行わないので注意してください。

5.7 トラブルシューティング：概要

この節では、OpenVMS オペレーティング・システムでの動作エラーの解釈における UETP の役割を説明します。UETP の実行時における通常のエラーおよびその修正方法については、5.8 項「トラブルシューティング：考えられる UETP エラー」を参照してください。

5.7.1 エラーの記録と診断

エラーが発生すると、UETP はユーザ・プログラムと同じように反応します。UETP はエラー・メッセージを返して継続するか、回復不可能なエラーを報告してイメージまたはフェーズを終了します。どちらの場合でも、UETP はハードウェアが適切に動作していることを想定しているので、エラーの診断は行いません。

エラーの原因が直ちに解明できない場合は、次の方法を使ってエラーを診断してください。

- OpenVMS Error Log Viewer (ELV) — エラー・ログ・ファイルを、ユーザが読める形式で、コマンド行から素早く調べることができます。データに対し、System Event Analyzer (SEA) などのツールによる、より分かりやすい分析が必要かどうかを判断できます。ELV の実行についての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』を参照してください。
- System Event Analyzer (SEA) — System Event Analyzer (SEA) は、エラー・イベントの分析と解釈を行う、ルール・ベースのハードウェア障害管理診断ツールです。SEA のマルチ・イベント相関分析機能により、システムのバイナリ・イベント・ログ・ファイルに格納されているイベントと、他のソースのイベントを分析することができます。SEA の使用についての詳細は、次の Web サイトを参照してください。

http://h18023.www1.hp.com/support/svctools/webes/sea_ug.pdf

- 診断機能 — 診断機能を使ってデバイスや媒体を徹底的にテストし、エラーの原因を特定する。

5.7.2 UETP 出力の中断

UETP テストの進行状況を、テストを起動したターミナルから監視することができます。このターミナルは、各フェーズの開始および終了時の通知およびエラーを知らせるメッセージなどの状態情報を常に表示します。

テストは、状態情報以外の出力をさまざまなログ・ファイルに送信します。ログ・ファイルの種類は、どのようにテストを起動したかによって異なります (5.7.7 項「ログ・ファイル」を参照)。ログ・ファイルには、テスト・プロシージャが作成した出力が入っています。UETP が正常終了し、ターミナルにエラーが表示されなかった場合でも、このようなログ・ファイルにエラーがないかチェックすることが大切です。さらに、ターミナルにエラーが表示された場合には、ログ・ファイルをチェックし、そのエラーの原因と性質の詳細を調べるようにしてください。

各テストは終了メールボックスを使って、最終完了状態をテスト・コントローラ・イメージ UETPHAS00 へ返します。この完了状態は、符号なしロングワード整数で、状態値を表します。トラブルシューティングの助けとなるように、UETPHAS00 は \$FAO および \$GETMSG システム・サービスを使って、テストの最終完了状態を表示します。

ただし、\$FAO サービスは、終了メールボックスを使っても提供できない追加の情報を必要とすることがあります。このような事態が発生すると、UETP は次のようなエラー・メッセージを表示します。

```
UETP-E-ABORT, !AS aborted at !%D
```

UETP がこのようなエラー・メッセージを表示するときは、ログ・ファイルをチェックし、詳細な情報を入手してください。個々のテストを実行し、問題を診断することも可能です。

ターミナルに表示されるエラー・メッセージ、およびログ・ファイルに格納されるエラー・メッセージは、基本的に次の 2 つから発行されたものです。

- UETP テスト
- テストされたシステム・コンポーネント

このメッセージが理解できないときには、OpenVMS ヘルプ・メッセージ・ユーティリティ (Help Message) を使用するか、『OpenVMS System Messages and Recovery Procedures Reference Manual』² または個々のシステム・コンポーネントについて記述されたマニュアルを参照してください。

5.7.3 画面に情報を表示する方法

デバイス・テスト UETINIT00.EXE, UETCLIG00.EXE, および UETDNET00.COM などのいくつかの UETP の部分では、テスト実行の進行に関する追加情報、またはテスト中に発生した問題に関する追加情報を入手できるものもあります。通常、この情報は重要ではないので、画面には表示されません。

この情報を見るためには、次のコマンドを入力して論理名 MODE を定義し、プログラムを実行します。

```
$ DEFINE MODE DUMP
```

5.7.4 画面表示の例 (VAX のみ)

次の例は、VAX 6000 コンピュータ上での UETINIT00.EXE の出力例で、論理名 MODE が DUMP と定義されている様子を示しています。

```
$ DEFINE MODE DUMP
$ RUN UETINIT00 (or @UETP)
```

```
Welcome to VAX/VMS UETP Version X7.3
```

```
%UETP-I-ABORTC, UETINIT00 to abort this test, type ^C
```

```
You are running on a VAX 6000-430 CPU with 327680 pages of memory.
The system was booted from _$11$DUA6:[SYS0.]
```

```
Run "ALL" UETP phases or a "SUBSET" [ALL]?
How many passes of UETP do you wish to run [1]?
```

```
The default number of loads is the minimum result of
```

```
1) CPU_SCALE * ( (MEM_FREE + MEM_MODIFY) / (WS_SIZE * PER_WS_INUSE) )
   7.32 * ( ( 232390 + 5048) / ( 1024 * 0.20) ) = 8486
2) Free process slots = 296
3) Free page file pages / Typical use of page file pages per process
   1099992 / 1000 = 1099
```

```
How many simulated user loads do you want [296]?
Do you want Long or Short report format [Long]?
```

```
UETP starting at 1-MAR-2001 16:00:43.86 with parameters:
DEVICE LOAD DECNET CLUSTER phases, 1 pass, 296 loads, long report.
```

2. このマニュアルはアーカイブ扱いです。

\$

このプログラムは、いかなるフェーズも起動しません。このプログラムは、ユーザ負荷および現在の実行中に使用する特定の要素を決定するのに UETP が使用する公式を表示します。

質問に Return を押して応答します。最初のプロンプトに応答した後、プログラムは同時プロセスの省略時の数を決定する公式を表示します。次の定義が適用されます。

- CPU_SCALE は、VAX 11/780 コンピュータに対する CPU の相対処理能力を示す。たとえば、VAX 11/780 を 1.0 とした場合、VAX 6000-430 コンピュータの CPU_SCALE は 7.32 である。これは、VAX 6000-430 が VAX 11/780 の 7.32 倍の処理能力を持つことを示す。
- MEM_FREE は、ユーザが利用できるメモリ (ページ単位) を表す。
- MEM_MODIFY は、修正されたページ・リスト上のメモリ・ページを表す。
- WS_SIZE は、ワーキングセット・サイズを表す。
- PER_WS_INUSE は、各プロセスでアクティブに使用されているワーキングセットの典型的なパーセンテージを表す。

また、UETINIT00 は公式が示す特定の値も表示します。上記の例では、UETP はシミュレートするユーザ負荷の省略時の値として 296 を選択しています。これは、296 は 3 つの公式の最低の結果だからです。

UETP の実行ごとにユーザ負荷の詳細を見るつもりがないのであれば、UETP の実行前に、論理名 MODE の割り当てを解除します。

5.7.5 画面表示の例 (Alpha および I64)

次の例は、Alpha システム上での UETINIT00.EXE の出力例で、論理名 MODE が DUMP と定義されている様子を示しています。

```
$ DEFINE MODE DUMP
$ RUN UETINIT00 (or @UETP)
```

```
Welcome to OpenVMS Alpha UETP Version 7.3
```

```
%UETP-I-ABORTC, UETINIT00 to abort this test, type ^C
```

```
You are running on a AlphaServer 4100 5/533 4MB CPU.
The system was booted from _$4$DKA300:[SYS0.]
```

```
Run "ALL" UETP phases or a "SUBSET" [ALL]?
How many passes of UETP do you wish to run [1]?
```

```
The default number of loads is the minimum result of
```

```
1) (MEM_FREE + MEM_MODIFY) / ( WS_SIZE )
   ( 1807872 +      10496) / ( 16512)           = 110
```

```
2) Free process slots                               = 488
```

```
3) Free page file pages / Typical use of blocks per process
   650240 /                                           1000 = 650
```

```
How many simulated user loads do you want [110]?
Do you want Long or Short report format [Long]?
```

```
UETP starting at 1-MAR-2001 15:53:19.52 with parameters:
DEVICE LOAD DECNET CLUSTER phases, 1 pass, 110 loads, long report.
```

このプログラムは、いかなるフェーズも起動しません。このプログラムは、ユーザ負荷および UETP が現在の実行中に使用する特定の要素を決定するのに使用される公式を表示します。

質問に Return を押して応答します。最初のプロンプトに応答した後、プログラムは同時プロセスの省略時の数を決定する公式を表示します。次の定義が適用されます。

- MEM_FREE は、ユーザが使用できるメモリ (ページレット単位) を示す。
- MEM_MODIFY は、修正されたページ・リスト上のメモリ・ページレットを表す。
- WS_SIZE は、ワーキングセット・サイズ (ページレット単位) を表す。

また、UETINIT00 は公式が示す特定の値も表示します。上記の例では、UETP はシミュレートするユーザ負荷の省略時の値として 110 を選択しています。これは、100 がこれら 3 つの公式の最低の結果だからです。

UETP の実行ごとにユーザ負荷の詳細を見るつもりがないのであれば、UETP の実行前に、論理名 MODE の割り当てを解除します。

5.7.6 UETP イーサネット・テスト用の遠隔ノードの定義

UETUNAS00 テスト中、障害報告がテスト中のデバイスに関連するのか、または遠隔デバイスに関連するのかを決定するのが困難な場合があります。適切なエラー報告を行うための最も簡単な方法は、**適切な転送場所**を定義することです。適切な転送場所というのは、イーサネット・パケットを正しく転送し、起動していて、レディ状態で待機しているということが判明している遠隔ノードのことです。

UETUNAS00 テストで、適切な転送場所を使用するためには、次のような操作を行います。次のコマンドでは、**適切な**デバイスがノード BETA 上にあり、そのノード BETA がすでにネットワーク・データベースで定義されているということを仮定しています。

1. ネットワーク制御プログラム (NCP) を使って、適切なイーサネット・ノードのアドレスを見つける。NCP を使用するには、次の条件を満たしていなければならない。

- DECnet for OpenVMS が起動し、システム上で動作していること
- 使用しているアカウントが、TMPMBX および NETMBX 特権を持っていること

次のコマンドを入力し、Return を押す。

```
$ RUN SYS$SYSTEM:NCP
NCP> TELL BETA SHOW EXECUTOR STATUS
```

ノード BETA がネットワーク・データベースで定義されていない場合、NCP はエラー・メッセージを表示する。この場合、他の適切なノードを指定し、もう一度上記コマンドを実行する。このノードが定義されている場合は、システム管理者またはネットワーク管理者に問い合わせる。

NCP は、次のような情報を表示する。

```
Node Volatile Status as of 22-JUN-2000 16:13:02
```

```
Executor node = 19.007 (BETA)
```

```
State = on
Physical address = AA-00-03-00-76-D3
Active links = 6
Delay = 1
```

2. 表示された *physical address* (この場合、AA00030076D3) を使って、論理名 TESTNIADR が適切な転送場所を指すように定義する。ハイフン (-) は指定しない。まず、SYSTEST アカウントにログインする。その後、次のコマンドを入力する。

```
$ DEFINE/SYSTEM TESTNIADR AA00030076D3
```

3. UETP を実行する。
4. UETP が完了したとき、次のコマンドを入力し、論理名 TESTNIADR の割り当てを解除する。

5.7.7 ログ・ファイル

UETP は、現在の実行中のすべての UETP テストおよびフェーズによって作成されたすべての情報を、1 つまたは複数の UETP.LOG ファイルに格納します。そして、前回の実行の情報を、1 つまたは複数の OLDUETP.LOG ファイルに格納します。UETP の実行が複数パスを呼び出す場合、パスごとに、UETP.LOG または OLDUETP.LOG ファイルが 1 つずつ作成されます。

実行を開始すると、UETP はすべての OLDUETP.LOG ファイルを削除し、すべての UETP.LOG ファイルの名前を、そのファイルのバージョンに相当する OLDUETP.LOG ファイルの名前に変更します。次に、UETP は新しい UETP.LOG ファイルを作成し、現在のパスの情報をその中に格納します。UETP のその後のパスでは、より高いバージョンの UETP.LOG が作成されます。したがって、複数パスを呼び出す UETP の実行終了時には、パスごとに、UETP.LOG ファイルが 1 つずつ作成されます。ファイル UETP.LOG および OLDUETP.LOG の作成にあたり、UETP は最新 2 回の実行からの出力を使用します。

クラスタ・テストは、実行に含まれる各システム上のパスごとに、NETSERVER.LOG ファイルを SYS\$TEST に作成します。テストがエラーを報告できない場合 (たとえば、他のノードへの接続が失われた場合)、そのノード上の NETSERVER.LOG ファイルに、そのノード上で実行されたテスト結果が格納されます。UETP は NETSERVER.LOG ファイルを削除またはパージしません。したがって、ときどき NETSERVER.LOG を削除して、ディスク領域を回復するようにしてください。

UETP の実行が正常終了しなかった場合、SYS\$TEST には他のログ・ファイルが格納されていません。通常、これらのファイルは連結され、UETP.LOG の中に格納されます。システム・ディスク上のログ・ファイルはすべてエラー・チェック用に使用できますが、新しいテストを実行する前には、これらのファイルをすべて削除しておかなければなりません。これらログ・ファイルはユーザが削除することもできますが、完全な UETP をもう一度実行すれば、古い UETP.LOG ファイルは自動的にチェックされ、削除されます。

5.8 トラブルシューティング：考えられる UETP エラー

この節では、UETP 実行時に発生する可能性のある問題の識別および解決に役に立つ情報を示します。システム障害を理解し、その原因を特定するとき、この節を参照してください。この節は、システムを回復したりユーザのシステムの欠陥を診断するためのマニュアルではありませんが、エラー・メッセージ中の情報を解釈し、それに対処する際に参考となります。

この節で述べる手順に従ってもエラーを回復できなかった場合は、弊社のサポート担当者に相談してください。このとき、問題を特定しようとして行った処置はすべてお知らせください。問題を診断する手掛かりになります。

5.8.1 一般的な障害の概要

次に、UETP の実行中に発生する最も一般的な障害を示します。

- クォータ、特権、アカウントの間違い
- UETINIT01 障害
- UETVECTOR 障害 (VAX コンピュータのみ)
- ディスク領域の不足
- クラスタ 設定の間違い
- ロード・テスト中の問題
- DECnet for OpenVMS のエラー
- 記録されるが表示されないエラー
- プロセス制御ブロック (PCB) または スワップ・スロットの欠如
- システム・ハングアップ
- ファイル・アクセス・リスナ (FAL) ・オブジェクトの省略時のアクセス権の欠如

- バグ・チェックおよびマシン・チェック

以降の項では、これらのエラーおよびその最善の対処方法を説明します。

5.8.2 クォータ, 特権, アカウントの間違い

割り当てたクォータまたは特権が SYSTEST アカウントの標準のクォータおよび特権と一致しない場合, UETP は次のエラー・メッセージを表示します。

```
*****
*   UETINIT00           *
*   Error count = 1    *
*****
-UETP-W-TEXT,   The following:

      OPER privilege,
      BIOLM quota,
      ENQLM quota,
      FILLM quota,
```

are nonstandard for the SYSTEST account and may result in UETP errors.

このメッセージは, OPER 特権, および BIOLM, ENQLM, FILLM の各クォータが正しく割り当てられていないか, または全く割り当てられていないことを示しています。



注意:

クラスタ統合テスト・フェーズを実行していて, SYSTEST_CLIG アカウントの特権およびクォータが間違っている場合, UETP はこのようなメッセージを表示します。SYSTEST および SYSTEST_CLIG アカウントには, 同じ特権およびクォータが必要です。どちらの場合も, ここで述べる対処方法を適用してください。

解決策

問題を修正するには, 次の手順に従ってください。

1. 次のように, Authorize ユーティリティ (AUTHORIZE) を使って, SYSTEST アカウントに対して有効なすべての特権およびクォータを表示する。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN SYS$SYSTEM:AUTHORIZE
UAF> SHOW SYSTEST

Username: SYSTEST                               Owner: SYSTEST-UETP
Account: SYSTEST                                UIC: [1,7] ([SYSTEST])
CLI: DCL                                         Tables: DCLTABLES
Default: SYS$SYSROOT:[SYSTEST]
LGICMD: LOGIN
Login Flags:
Primary days: Mon Tue Wed Thu Fri Sat Sun
Secondary days:
No access restrictions
Expiration: (none)                               Pwdminimum: 8 Login Fails: 0
Pwdlifetime: 14 00:00                             Pwdchange: 22-JUN-2000 10:12
Last Login: (none) (interactive), (none) (non-interactive)
Maxjobs: 0 Fillm: 100 Byt1m: 65536
Maxacctjobs: 0 Shrfillm: 0 Pbyt1m: 0
Maxdetach: 0 BIOLm: 12 JTquota: 1024
Prclm: 12 DIOLm: 55 WSdef: 256
Prio: 4 AST1m: 100 WSquo: 512
Queprio: 0 TQElm: 20 WSextent: 2048
CPU: (none) Enqlm: 300 Pgflquo: 20480
Authorized Privileges:
CMKRNL CMEXEC SYSNAM GRPNAM DETACH DIAGNOSE LOG_IO GROUP
PRMCEB PRMMBX SETPRV TMPMBX NETMBX VOLPRO PHY_IO SYSPRV
Default Privileges:
CMKRNL CMEXEC SYSNAM GRPNAM DETACH DIAGNOSE LOG_IO GROUP
```

```
PRMCEB PRMMBX SETPRV TMPMBX NETMBX VOLPRO PHY_IO SYSPRV
UAF> SHOW SYSTEST_CLIG
```

```
UAF> EXIT
```

2. このアカウントに割り当てられた省略時の特権およびクォータが、次の値と一致しているかどうか確認する。

特権

CMKRNL	CMEXEC	NETMBX	DIAGNOSE IMPERSONATE
DETACH	PRMCEB	PRMMBX	PHY_IO
GRPNAM	TMPMBX	VOLPRO	LOG_IO
SYSNAM	SYSPRV	SETPRV	GROUP

クォータ

BIOLM: 150	PRCLM: 8
DIOLM: 150	ASTLM: 250
FILLM: 100	BYTLM: 64000
TQELM: 20	CPU: 制限なし
ENQLM: 2000	PGFLQUOTA: 50000 (Alpha - 800,000)
WSDEFAULT: 2000	WSQUOTA: 4000
WSEXTENT: 16384 (16)	

3. 特権またはクォータのいずれかが間違っている場合、AUTHORIZE を実行して修正する。間違ったアカウントにログインした場合は、次のエラー・メッセージが表示され、SYSTEST アカウントにログインするかどうか尋ねられます。

```
$ @UETP
```

```
*****
* UETINIT00 *
* Error count = 1 *
*****
-UETP-E-ABORT, UETINIT00 aborted at 22-JUN-2000 14:24:10.13
-UETP-E-TEXT, You are logged in to the wrong account.
Please log in to the SYSTEST account.
```

```
$
```

UETP は SYSTEST アカウントから実行しなければなりません。

5.8.3 UETINIT01 障害

UETINIT01 障害は、周辺機器デバイスに関連します。このタイプのエラー・メッセージは、次のいずれかを示します。

- デバイス障害
- デバイスがサポートされていない、またはマウントされていない。
- デバイスが他のユーザに割り当てられている。
- デバイスに書き込みロックがかけられている。
- 磁気テープ・ドライブに空白がない。
- ドライブがオフラインである。

エラー・メッセージの中には、その修正方法が示されているものもあります。たとえば、問題および推奨する修正方法を明示的に知らせるメッセージを、オペレータ通信マネージャ (OPCOM) から受け取ることがあります。

```
%OPCOM, 22-JUN-2004 14:10:52.96, request 1, from user SYSTEST
Please mount volume UETP in device _MTA0:
%MOUNT-I-OPRQST, Please mount volume UETP in device _MTA0:
```

解決方法が暗黙に示されているメッセージもあります。

```
%UETP-S-BEGIN, UETDISK00 beginning at 22-JUN-2004 13:34:46.03
```

```
*****
*   DISK_DRA           *
*   Error count = 1   *
*****
```

```
-UETP-E-TEXT, RMS file error in file DRA0:DRA00.TST
-RMS-E-DNR, device not ready or not mounted
%UETP-S-ENDED, UETDISK00 ended at 22-JUN-2004 13:34:46.80
```

このメッセージは、ディスク・ドライブがレディ状態でないか、マウントされていないことを示しています。この情報から、(ディスク・ドライブの)どの場所に障害の原因があるかを知ることができます。即座に問題の原因を知ることができない場合は、5.3 項「テストを行うデバイスの設定」の設定指示を参照してください。

また、障害の原因の手掛かりが全くないメッセージもあります。問題は、ソフトウェアでなく、ハードウェアに原因があることもあります。

解決策

UETP の実行中、いつまたはどこで障害が発生したかを判断するには、次の手順に従ってください。

- デバイス・テストを個々に実行する (5.5.1 項「フェーズのサブセットの実行方法」を参照)。こうすることによって、障害の再現性を調べることができる。さらに、最少のソフトウェアを使って問題を再現するので、問題の原因を特定することができる。
たとえば、完全なデバイス・フェーズを実行するときだけ障害が発生し、関連するデバイスに対して個別にテストを実行するときは障害が発生しない場合、この問題の原因はデバイスの相互作用であると推測することができる。逆に、1つのデバイス・テストを実行するときにエラーが再現する場合は、そのエラーの原因はデバイスの相互作用には関係ないと推測することができる。
- 異なる媒体でデバイス・テストを実行する。1つのデバイス・テストを実行するときにエラーが再現する場合、磁気テープまたはディスク・メディアに欠陥がある可能性がある。同じテストを異なる媒体で実行すれば、問題の原因が媒体であるかどうかを決定できる。
- 上記の手順でもまだ問題が解決できない場合は、弊社のサポート担当者に相談する。

5.8.4 UETVECTOR 障害 (VAX のみ)

UETP は、次のようなメッセージを表示して、ベクタ・プロセッサ障害を通知します。

```
*****
*   UETVECTOR         *
*   Error count = 1   *
*****
%PPL-S-CREATED_SOME, created some of those requested - partial success
-UETP-E-SUBSPNERR, Error spawning subordinate process.
-UETP-E-SCHCTXERR, Error scheduling vector context test subprocess.
-UETP-E-VECCTXERR, Error encountered during vector context testing.
%UETP-I-ENDED, UETVECTOR_0000 ended at 22-JUN-2004 07:37:00.59
```

解決策

ベクタ・プロセッサのテストのための正しい設定については、5.3.19 項「ベクタ・プロセッサおよび VVIEF (VAX のみ)」を参照してください。

5.8.5 ディスク領域の不足

UETP のパスを連続して実行すると、UETP を実行したディスク上にログ・ファイルが蓄積されます。これらのファイルによって、後続のパスで使用できる空きディスク領域が減少します。現在のロードに対して使用できるディスク領域が少なくなった場合は、次のエラー・メッセージが表示されます。

```
%UETP-S-BEGIN, UETDISK00 beginning at 22-JUN-2004 08:12:24.34
%UETP-I-ABORTC, DISK_DJA to abort this test, type ^C

*****
* DISK_DJA *
* Error count = 1 *
*****
-UETP-F-TEXT, RMS file error in file DJA0:DJA00.TST
-RMS-F-FUL, device full (insufficient space for allocation)

*****
* DISK_DJA *
* Error count = 2 *
*****
-UETP-F-TEXT, RMS file error in file DJA0:DJA01.TST
-RMS-F-FUL, device full (insufficient space for allocation)
%UETP-E-DESTP, DISK_DJA stopped testing DJA unit 0 at 08:12:36.91
%UETP-S-ENDED, UETDISK00 ended at 22-JUN-2004 08:12:37.98
```

解決策

ディスク上の使用できる領域を増やしてください。領域を増やすには、次に挙げる 1 つまたは複数の方法を使用します。

- 不必要なファイルを削除し、より多くの領域を作成する。
- 複数のバージョンが存在する場合、ファイルをパージする。
- 十分な領域を持つボリュームをマウントする。
- ディスク・クォータがディスク上に設定されているかどうかチェックする。設定されている場合、ディスク・クォータを使用不可にするか、増やす(ディスク・クォータ・ユーティリティについては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照)。
- 小規模ディスク・システムの場合、VMSTAILOR を実行する。詳細は、オペレーティング・システムのアップグレードおよびインストール・マニュアルを参照。

ディスク領域についての詳細は、5.2.2 項「SYSTEST ディレクトリの使用方法」および 5.3.3 項「UETP のディスク上での動作」を参照してください。

5.8.6 OpenVMS Cluster システムの設定の間違い

クラスタ統合テスト中に発生する問題のほとんどの原因は、OpenVMS Cluster か OpenVMS Cluster 上の UETP の設定の誤りです。これらの問題のほとんどは、クラスタ・テストの次の段階で発生するようです。

- 開始直後、OpenVMS ノード上のプロセスが起動されたとき
- 終了直前、クラスタ・ファイル・アクセスがチェックされたとき

クラスタ・テスト・フェーズでは、ユーザのクラスタ中のさまざまな OpenVMS ノードが、クラスタ内の選択したノード上のファイルに同時にアクセスしている様子を見ることができます。UETP は、最初に、クラスタ内の選択した他のノードからアクセス可能なディスク・ドライブ上にファイルを作成しようとします。クラスタ・テスト・フェーズ中にファイルを作成するための要件を次に示します。

- マスタ・ファイル・ディレクトリ (MFD) またはルート・ディレクトリ [SYS0.] のいずれかのディスク上に、[SYSTEST] ディレクトリが存在しなければならない。

- [SYSTEST] ディレクトリに対する保護は、SYSTEST アカウントがそのディレクトリの中にファイルを作成できるように設定されていなければならない。

UETP がどこかのノード上に適切なデバイスを見つけることができなかった場合は、警告メッセージが表示され、次のクラスタ・ノードに進みます。

オペレータのターミナル (OPA0) に NO BROADCAST ターミナル特性が設定されているノードでは、クラスタ・テスト中に次のエラー・メッセージが表示されます。

```
*****
* UETCLIG00master *
* Error count = 1 *
*****
-UETP-E-TEXT, 0 operator consoles timed out on the cluster test warning
and 1 operator console rejected it.
-UETP-E-TEXT, Status returned was,
"%SYSTEM-F-DEVOFFLINE, device is not in configuration or not
available"
```

OPA0 に NO BROADCAST が設定されていない場合は、このメッセージを無視してください。

解決策

問題の疑いがある場合は、SYSTEST_CLIG プロセスが作成されたときに作成された SYS\$TEST:NETSERVER.LOG ファイルを調べてください。このファイルには、テストを実行しているノードに転送できなかった追加のエラー情報が入っていることもあります。いくつかのノード上で SYSTEST_CLIG プロセスを作成できなかった場合、そのノードに対するシステム会計情報ファイルのプロセス終了レコードには、最後のプロセス状態が入っていることがあります。

次の問題は、クラスタ・テスト中に発生する可能性があるものです。

- 他のノードへログインする — この問題の原因は、クラスタ・テストのための遠隔 OpenVMS ノードの設定が間違っていることである。たとえば、SYSTEST_CLIG アカウントのパスワードを指定したり、SYSTEST_CLIG アカウントを使用不可にしている場合、テストは次のメッセージを表示する。

```
%SYSTEM-F-INVLOGIN, login information invalid at remote node
```

クラスタ・テストのための準備についての詳細は、5.3.15 項「OpenVMS Cluster のテスト」および 5.7.6 項「UETP イーサネット・テスト用の遠隔ノードの定義」を参照。

- 他のノードと通信する — このメッセージは DECnet の問題であることを示している。影響のあったノード上の NETSERVER.LOG ファイルをチェックし、原因を決定すること。
- ロックを外すまたはデッドロックを検出しない — ほとんどの場合、この問題の原因は、SYSTEST アカウントにログインしていないことである。他には、ユーザのクラスタが適切に構成されていないことが考えられる。
- クラスタ・ノード上へファイルを作成する — この問題の原因は、クラスタ・テストのための設定の間違いである。クラスタ・テストのために準備については、5.3.15 項「OpenVMS Cluster のテスト」を参照。

5.8.7 ロード・テスト中の問題

ロード・テスト中にはさまざまなエラーが発生します。これは、テスト中に起動されるコマンド・プロシージャは、複数のユーティリティを実行し、さまざまな機能を行うからです。UETP はロード・テスト中に作成したログ・ファイルを削除するので、問題の追跡が困難なことがあります (5.9.3 項「システム・ロード・テスト・フェーズ」を参照してください)。

解決策

ロード・テスト中に問題が発生し、その原因が分からない場合は、次のように UETP.COM を変更して、ログ・ファイルを保持するようにします。

1. 次の行に /NODELETE 修飾子を追加する。

```
$ TCNTRL UETLOAD00.DAT/PARALLEL_COUNT='LOADS/REPORT_TYPE='REPORT
```

2. 次の行を削除するか、コメントにする。

```
$ DELETE UETLO*.LOG;*
```

変更後、もう一度ロード・テストを行い、問題が再現するかどうか確かめます。

問題が再現する場合は、適切なログ・ファイルの内容を調べます。どのログ・ファイルを読むべきかを判断するには、ロード・テストがそのプロセスとログ・ファイルに名前を付けた流れを理解します (ログ・ファイル名はプロセス名を継承します)。

ロード・テストは、作成したプロセスに、次の形式の名前を付けます。

UETLOAD_{nn}__{nnnn}

次に例を示します。

```
%UETP-I-BEGIN, UETLOAD00 beginning at 22-JUN-2004 15:45:08.97
%UETP-I-BEGIN, UETLOAD02_0000 beginning at 22-JUN-2004 15:45:09.42
%UETP-I-BEGIN, UETLOAD03_0001 beginning at 22-JUN-2004 15:45:09.63
%UETP-I-BEGIN, UETLOAD04_0002 beginning at 22-JUN-2004 15:45:10.76
%UETP-I-BEGIN, UETLOAD05_0003 beginning at 22-JUN-2004 15:45:11.28
%UETP-I-BEGIN, UETLOAD06_0004 beginning at 22-JUN-2004 15:45:12.56
%UETP-I-BEGIN, UETLOAD07_0005 beginning at 22-JUN-2004 15:45:13.81
%UETP-I-BEGIN, UETLOAD08_0006 beginning at 22-JUN-2004 15:45:14.95
%UETP-I-BEGIN, UETLOAD09_0007 beginning at 22-JUN-2004 15:45:16.99
%UETP-I-BEGIN, UETLOAD10_0008 beginning at 22-JUN-2004 15:45:19.32
%UETP-I-BEGIN, UETLOAD11_0009 beginning at 22-JUN-2004 15:45:19.95
%UETP-I-BEGIN, UETLOAD02_0010 beginning at 22-JUN-2004 15:45:20.20
%UETP-I-BEGIN, UETLOAD03_0011 beginning at 22-JUN-2004 15:45:21.95
%UETP-I-BEGIN, UETLOAD04_0012 beginning at 22-JUN-2004 15:45:22.99
```

10 以上のプロセスが作成されている場合、プロセス名の UETLOAD_{nn} 部分の連番は、UETLOAD02 から始まります。しかし、_{nnnn} 部分の 4 けたの数字は、そのまま増え続けます。

ロード・テストのプロセスごとに、2 つのログ・ファイルが作成されます。最初のログ・ファイルは、テスト・コントローラによって作成されます。2 番目のログ・ファイルは、そのプロセス自身によって作成されます。ロード・テストのプロセスについてのエラー情報を調べるときには、テスト・コントローラが作成したログ・ファイル (最初のログ・ファイル) を調べます。

ロード・テストのログ・ファイルの名前はプロセス名を継承し、UETLO にプロセス名の最後の 4 けたの数字 (_{nnnn} 部分) を追加します。各プロセスのテスト・コントローラのログ・ファイルおよびプロセスのログ・ファイルは同じファイル名です。ただし、プロセスのログ・ファイルの方が、より高いバージョン番号を持っています。たとえば、プロセス UETLOAD05_0003 が作成したログ・ファイルの名前は次のようになります。

UETLO0003.LOG;1 (テスト・コントローラのログ・ファイル)

UETLO0003.LOG;2 (プロセスのログ・ファイル)

ログ・ファイルを見るときには、ロード・テストのコマンドおよびエラー情報の入った、バージョン番号の低いほうを見るようにしてください。

問題を解決したら、UETP.COM を元の状態に戻し、ロード・テストのログ・ファイルを削除します (UETLO*.LOG;*)。このファイルを削除しなければ、ディスク領域の問題が発生する可能性があります。

5.8.8 DECnet for OpenVMS エラー

DECnet エラー・メッセージは、ネットワークが使用不可であることを示すことがあります。

解決策

- DECnet for OpenVMS ソフトウェアがユーザのシステムの中にある場合は、次のコマンドを入力して、製品登録キー (PAK) が登録されているかどうか確かめること。

```
$ SHOW LICENSE
```

PAK が登録されていない場合、ライセンス・ユーティリティを呼び出し、次のコマンドを入力して、登録を行うこと。

```
$ @SYS$UPDATE:VMSLICENSE
```

ライセンスの登録については、以下を参照。

- ユーザのオペレーティング・システムのアップグレードおよびインストール・マニュアル
 - 『OpenVMS License Management Utility Manual』
- DECnet for OpenVMS ソフトウェアがユーザのシステムにない場合、このメッセージを無視する。これは通常の状態であり、UETP の実行には影響しない。

他の DECnet に関するエラーが発生した場合は、次の作業を行ってください。

- DECnet for OpenVMS ソフトウェアを 1 つのフェーズとして実行し (5.5.1 項「フェーズのサブセットの実行方法」を参照)、そのエラーに再現性があるかどうかを決定する。
- Help Message を使用する。または『OpenVMS System Messages: Companion Guide for Help Message Users』を参照する。

5.8.9 記録されるが表示されないエラー

コンソール・ターミナルにエラーが表示されない場合、または UETP.LOG ファイルにエラーが報告されない場合、Error Log Viewer (ELV) を実行して、ERRLOG.SYS ファイルになんらかのエラーが記録されているかどうかを確認します。ELV の実行については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』を参照してください。

5.8.10 PCB またはスワップ・スロットの欠如

次のエラー・メッセージは、PCB またはスワップ・スロットが使用できないことを示しています。

```
%UETP-I-BEGIN, UETLOAD00 beginning at 22-JUN-2004 07:47:16.50
%UETP-I-BEGIN, UETLOAD02_0000 beginning at 22-JUN-2004 07:47:16.76
%UETP-I-BEGIN, UETLOAD03_0001 beginning at 22-JUN-2004 07:47:16.92
%UETP-I-BEGIN, UETLOAD04_0002 beginning at 22-JUN-2004 07:47:17.13
%UETP-I-BEGIN, UETLOAD05_0003 beginning at 22-JUN-2004 07:47:17.35
%UETP-I-BEGIN, UETLOAD06_0004 beginning at 22-JUN-2004 07:47:17.61
%UETP-W-TEXT, The process -UETLOAD07_0005- was unable to be created,
the error message is
-SYSTEM-F-NOSLOT, no pcb or swap slot available
%UETP-W-TEXT, The process -UETLOAD08_0006- was unable to be created,
the error message is
-SYSTEM-F-NOSLOT, no pcb or swap slot available
%UETP-W-TEXT, The process -UETLOAD09_0007- was unable to be created,
the error message is
-SYSTEM-F-NOSLOT, no pcb or swap slot available
%UETP-W-TEXT, The process -UETLOAD10_0008- was unable to be created,
the error message is
-SYSTEM-F-NOSLOT, no pcb or swap slot available
%UETP-W-TEXT, The process -UETLOAD11_0009- was unable to be created,
the error message is
-SYSTEM-F-NOSLOT, no pcb or swap slot available
%UETP-W-ABORT, UETLOAD00 aborted at 22-JUN-2004 07:47:54.10
-UETP-W-TEXT, Aborted via a user Ctrl/C.
*****
```

```
*
END OF UETP PASS 1 AT 22-JUN-2004 07:48:03.17
*
*****
```

解決策

この問題を解決するには、次の手順に従います。

1. エラー・メッセージの原因となったフェーズを個々に実行し (上記例では LOAD フェーズ)、エラーが再現するか確認する。
2. コマンド・プロシージャ SYS\$UPDATE:SWAPFILES.COM (第2章「ページ・ファイル、スワップ・ファイル、ダンプ・ファイルの管理」を参照) または SYSGEN (『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照) を使って、ページ・ファイルのサイズを増やす。
3. 必要であれば、システム・パラメータ MAXPROCESSCNT を増やす。
4. システムをリブートする。

5.8.11 キーボードの応答がない、またはシステム・ディスクが動作しない

キーボードの応答がなかったり、システム・ディスクが動作していない場合、システムがハングアップしている恐れがあります。

解決策

システム・ハングアップの場合、障害追跡が困難です。参照のために、ダンプ・ファイルをセーブしておいてください。システムがハングアップした原因を特定するためには、『OpenVMS VAX System Dump Analyzer Utility Manual』と『OpenVMS Alpha System Analysis Tools Manual』のシステム・ダンプ・アナライザを参照してください。

システム・ハングアップの理由には、次のようなものがあります。

- プール領域の不足 — システム・パラメータ NPAGEVIR の値を増やし、システムをリブートする。
- ページ・ファイル領域の不足 — SYSGEN を使って、ページ・ファイル領域を増やす。『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』を参照。
- ドライバの無限ループによる I/O デバイス障害 — 弊社のサポート担当者に連絡する。

5.8.12 FAL オブジェクトに対する省略時のアクセス権の欠如

UETP の DECnet テストにより選択された遠隔ノード (アクティブな各サーキット上の隣接ノード、またはグループ論理名 UETP\$NODE_ADDRESS で定義されたノード) で、省略時の FAL アクセスが無効になっている場合は、次のようなメッセージが表示されます。

```
%UETP-W-TEXT, The process -SVA019841_0001- returned a final status of:
%COPY-E-OPENOUT, error opening !AS as output
```

上記メッセージは次のように続きます。

```
%COPY-E-OPENOUT, error opening 9999""::SVA019841.D1; as output
-RMS-E-CRE, ACP file create failed
-SYSTEM-F-INVLOGIN, login information invalid at remote node
%COPY-W-NOTCOPIED, SYS$COMMON:[SYSTEST]UETP.COM;2 not copied
%UETP-E-TEXT, Remote file test data error
```

このメッセージは無視して構いません。

5.8.13 バグ・チェックおよびマシン・チェック

システムがその実行を強制終了するとき、バグ・チェック・メッセージがコンソールに表示されます。

解決策

弊社のサポート担当者に連絡してください。バグ・チェックやマシン・チェックの原因は、たいていはハードウェア障害ですが、バグ・チェックやマシン・チェックは、簡単には解決できません。ただし、検査に使えるよう、SYS\$SYSTEM:SYSDUMP.DMP ファイルと ERRLOG.SYS ファイルを保存しておくことが重要です。この障害が再現できるかどうかも確認しておいてください。もう一度 UETP を実行して、障害をチェックすることができます。

5.9 UETP テストおよびフェーズ

この節では、UETP の編成およびテスト・パッケージの個々の構成要素を詳しく説明します。UETP を実行するには、各テスト・フェーズを起動するコマンドの入ったマスタ・コマンド・プロシージャを起動します。このプロシージャは、開始時に、さまざまなテスト・フェーズに必要な情報を入力するよう求めてきます (UETP の起動についての詳細は、5.5 項「UETP の起動」を参照してください)。

マスタ・コマンド・プロシージャ UETP.COM には、各テスト・フェーズを起動するコマンドが入っています。また、UETP.COM には、論理名の定義やテストによって作成されるファイルの操作などを行うコマンドも入っています。

UETP.COM プロシージャは、順番に各テスト・フェーズを制御する、テスト制御プログラム UETPHAS00.EXE を起動するコマンドを発行します。このテスト・コントローラは、複数の独立プロセスを起動します。さらに、これら独立プロセスの完了状態を報告し、独立プロセスが報告した情報も報告します。

以降の項では、さまざまな UETP テスト・フェーズについて説明します。

5.9.1 初期化フェーズ

初期化フェーズでは、次のことが行われます。

- イメージ UETINIT00.EXE が情報を求めるプロンプトを表示する (5.5 項「UETP の起動」を参照)。ここで入力した情報は、UETP テストの実行に影響する変数を定義する。
- イメージ UETINIT01.EXE は、システム中のすべてのコントローラおよび関連するデバイスの情報を収集する。このイメージは、UETINIDEV.DAT と呼ばれるファイルに情報を書き込む。
- UETINIT01.EXE は、UETSUPDEV.DAT 中の情報を使って、適切なデバイス・テストの実行によって操作可能なデバイスを確認する。各デバイス・テストでは、デバイスごとに簡単な読み込み操作と書き込み操作が行われる。デバイスがこのテストに失敗すると、UETINIDEV.DAT 中のそのデバイスのエントリは、そのデバイスがテストできないことを書き込む。後続の UETP テストはそのデバイスを無視する。
- テスト可能な各コントローラに対して、UETINIT01.EXE は UETCONT00.DAT と呼ばれるファイルに行を書き込む。この行は、テスト・ファイルをテストするコントローラに関連付ける。

UETINIDEV.DAT の要約は常に UETP.LOG 中に存在します。そして、長いレポート形式が要求されたときに、UETINIT01.EXE はこの要約をコンソールに送ります。

5.9.2 デバイス・テスト・フェーズ

デバイス・テスト・フェーズには、ディスク、磁気テープ、ライン・プリンタ、およびターミナルなどのデバイスのタイプ別のテストが含まれます。この項では、デバイス・テスト・フェーズについて説明し、単一のデバイスをテストするための指示を示します。デバイス・テスト・フェーズ全体を独立させて実行するための情報については、5.5.1 項「フェーズのサブセットの実行方法」を参照してください。

5.9.2.1 デバイス・フェーズの動作

UETP デバイス・テスト・フェーズは、実行可能イメージであるフェーズ・コントローラ UETPHAS00 を起動します。このイメージは、テストを行うデバイス・コントローラごとに、

独立プロセスを1つずつ作成します。たとえば、システムに3つのターミナル・コントローラ、1つのライン・プリンタ、および2つのディスク・コントローラが存在する場合、このイメージは6つの独立プロセスを作成します。同時に、独立プロセスは、さまざまなタイプのデバイスをテストするイメージを実行します。

UETPの初期化フェーズでは、UETINIDEV.DATと呼ばれるファイルとUETCONT00.DATと呼ばれるファイルが作成されます。UETINIDEV.DATには、OpenVMSがサポートするシステム中のコントローラおよび関連デバイスに関するデータが入っています。UETCONT00.DATは、デバイス・テスト・イメージをテスト可能な各コントローラに関連付けます。

UETPHAS00はUETCONT00.DAT中の情報を使って、作成した各デバイス・プロセスに渡すデバイス・コントローラ名を見つけます。UETPHAS00は、個々のテストへのSYS\$INPUTであるメールボックスにコントローラ名を書き込みます。各独立プロセスはそのデータを使って、テストを行うコントローラを決定します。次に、テスト・イメージはUETINIDEV.DATからデバイス・コントローラおよびそのコントローラ上でテスト可能なすべてのユニットを探します。すべてのコントローラ上のすべてのデバイスのテストが完了すると、フェーズ・コントローラは終了します。

UETCONT00.DATはUETPの実行終了時に自動的に削除されるので、UETP.COMを起動しなければデバイス・フェーズは実行できません。しかし、個々のテスト・イメージだけなら実行できます。UETINIDEV.DATは、ユーザが削除するまでSYS\$TESTに存在します。

5.9.2.2 単一デバイス・テストの実行

この項で述べる個々のテストを実行するには、まず、SYSTESTアカウントにログインしなければなりません。また、UETINIDEV.DATのコピーが存在しなければなりません。前回の実行によるこのファイルのコピーが存在しなければ、このファイルを作成してください(UETINIDEV.DATは、完全なUETPの実行またはデバイス・テスト・フェーズの実行によって作成されます)。単一デバイス・テストを実行するときには、ログ・ファイルは作成されないので注意してください。このテストはすべての出力をユーザのターミナルに送ります。

特定のデバイス・タイプに対してだけテストを行う場合は、表 5-1 「デバイス・テスト (VAXのみ)」 (VAXシステムの場合) または表 5-2 「デバイス・テスト (Alphaのみ)」 (Alphaシステムおよび164システムの場合) からテスト・イメージ名を選択し、そのイメージを実行することによって、特定のコントローラだけをテストすることができます。次に例を示します。

```
$ RUN UETTTYS00
Controller designation?: TTB
```

UETPは、コントローラ指示子およびデバイス・コードの入力を求めます。ユーザ自身のターミナルをテストするのでなければ、コントローラ名を明示的に示さなければなりません。ターミナル・テストを実行する場合、Returnを押せば、ユーザ自身のターミナルだけをテストできます。

何回も実行を繰り返す場合は、論理名CTRLNAMEを定義する方が便利です。

```
$ DEFINE CTRLNAME TTB
$ RUN UETTTYS00
```

このようにコントローラ名を定義すると、テスト完了後も論理名CTRLNAMEは割り当てられたままになります。論理名の割り当てを解除するには、次のように、DCLのDEASSIGNコマンドを使用します。

```
$ DEASSIGN CTRLNAME
```

5.9.2.3 UETINIDEV.DATの形式

UETINIDEV.DATファイルはASCII順編成ファイルなので、必要に応じて入力したり編集したりすることができます。このファイルの内容は、次のコマンドを入力することによって表示することができます。

```
$ TYPE UETINIDEV.DAT
```

```
DDB x ddd  
UCB y uuuuu nnnnnnnnnn.nnn  
END OF UETINIDEV.DAT
```

この例では、次のようにシンボルが定義されています。

シンボル	値
x	当該コントローラにテスト可能なユニットがある場合は T。当該コントローラでテストを行わない場合は N。
y	ユニットがテスト可能な場合は T。ユニットがテスト可能でない場合は N。
ddd	デバイス・コントローラ名。たとえば、DUA。
uuuuu	デバイス・ユニット番号。たとえば、25。
nnnnnnnnn.nnn	ユニットに対する UETP デバイス・テスト名。たとえば、UETDISK00.EXE。

UETINIDEV.DAT には、ユーザのシステムに接続されている、またはユーザのシステムから見る
ことができる各コントローラに対する DDB (デバイス・データ・ブロック) 行が入っています。
DDB 行の後には、当該コントローラに接続された各ユニットに対する UCB (ユニット制御ブ
ロック) 行があります。DDB 行と UCB 行の両方がそのデバイスがテスト可能であることを示
すときに限り、デバイス・テストは特定のデバイスをテストできます。

5.9.2.4 ループ・モードによるテストの実行

あるデバイスに対して特に負荷をかけてテストしたい場合は、ループ・モードでデバイス・テ
ストを実行します。このモードでは、テストが無限に実行されます。次に例を示します。

```
$ DEFINE MODE LOOP  
$ RUN UETDISK00  
Controller designation?: DRA  
%UETP-I-TEXT, End of pass 1 with 980 iterations at 22-JUN-2004 16:18:51:03
```

^c

ループ・モードのテストを終了するには、Ctrl/C を使用しなければなりません。Ctrl/Y を使
用すると、UETP はクリーンアップ処理を行いません。

5.9.2.5 個々のデバイス・テストの機能

ディスク・テストは、システム中のディスクごとにファイルを 2 つずつ割り当て、そのファイ
ルにランダムなデータ・ブロックを書き込みます。次に、テストはそのデータをチェック
し、エラーがあれば SYS\$OUTPUT に報告し、最後に、そのディスク・ファイルを削除します。

クラスタ環境でディスク・テスト・フェーズを実行する場合、テストを行うシステムにマウン
トされているすべてのディスクがアクセスされます。このとき、テストされるディスクのユー
ザにとってディスク領域の不足という問題が発生するかもしれません。したがって、遠隔ノ
ード上のユーザ(ローカル・システムのユーザとディスクを共有しているユーザ)には、使用中
のディスクが UETP によってテストされるということを警告しておくべきです。

磁気テープ・テストは、システム中のすべての磁気テープ・ドライブをテストします。この
テストは、マウントされているすべての磁気テープ上に大きなファイルを作成し、その中にさ
まざまなサイズの順次レコードを複数書き込みます。さらに、レコードの書き込み後、磁気
テープを巻き取り、書き込んだレコードを検査し、最後に、磁気テープを再初期化します。

ターミナルおよびライン・プリンタ・テストでは、数ページまたは数画面かの出力が作成さ
れ、各ページまたは画面にヘッダ行および ASCII 文字によるテスト・パターンが出力されま
す。ヘッダ行には、テスト名、デバイス名、データ、および時間が出力されます。

実験周辺機器アクセラレータ (LPA11-K) の場合、テスト・イメージによって LPA11-K の I/O バスの構成が決定されます。テスト・イメージはすべてのタイプのマイクロコードを LPA11-K にロードし、LPA-K の I/O バス上のすべてのデバイスに対してデータの読み込みまたは書き込みを行います。

通信デバイス・テストは、転送メッセージ・バッファをランダムなデータでいっぱいにします。次に、ループバック・モードを使って、メッセージを何度か転送および受信します。ループ・バックされたデータが正しいかどうかをチェックするために、AST ルーチンが \$QIO 読み込みと関連付けられ、受信したメッセージと転送したメッセージが比較されます。この手順は、異なる長さのメッセージを使って繰り返されます。

インタフェース・デバイス・テストは、デバイスを保守モードでテストし、ランダムなデータを書き込み、最後にそのデータを検査します。

ベクタ・プロセッサ・デバイス・テストは、単純なベクタ・スカラー算術演算およびベクタ・ベクタ算術演算を行い、その結果と予期された値を比較します。このテストは、また、ベクタ関連拡張システム・サービスを使って、強制的に算術例外条件およびメモリ管理例外条件をシステムに発生させます。

表 5-1 「デバイス・テスト (VAX のみ)」に、デバイス・テスト・イメージおよび VAX システム上でテストされるデバイスのリストを示します。

表 5-1 デバイス・テスト (VAX のみ)

テスト・イメージ名	テストされるデバイス
UETDISK00.EXE	ディスク
UETTAPE00.EXE	磁気テープ・ドライブおよびテープ・カートリッジ・ドライブ
UETTTYS00.EXE	ターミナルおよびライン・プリンタ
UETLPAK00.EXE	LPA11-K
UETCOMS00.EXE	DMC11, DMR11
UETDMPF00.EXE	DMF32, DMP11
UETDR1W00.EXE	DR11-W
UETDR7800.EXE	DR780, DR750
UETCDRO00.EXE	RRD40, RRD42, RRD50
UETUNAS00.EXE	イーサネット・アダプタ
UETVECTOR.EXE	ベクタ・プロセッサ, VVIEF

表 5-2 「デバイス・テスト (Alpha のみ)」に、デバイス・テスト・イメージおよび Alpha 上でテストされるデバイスのリストを示します。

表 5-2 デバイス・テスト (Alpha のみ)

テスト・イメージ名	テストされるデバイス
UETDISK00.EXE	ディスク
UETTAPE00.EXE	磁気テープ・ドライブおよびテープ・カートリッジ・ドライブ
UETTTYS00.EXE	ターミナルおよびライン・プリンタ
UETCDRO00.EXE	RRD42
UETUNAS00.EXE	イーサネット・アダプタ

5.9.3 システム・ロード・テスト・フェーズ

システム・ロード・テストの目的は、システム資源を同時に要求する複数のターミナル・ユーザをシミュレートすることです。システム・ロード・テストは、さまざまなコマンド・プロ

シー ज्याを実行する独立プロセスを複数作成します。この結果は、ファイル UETLOAD00.DAT に出力されます。各プロセスは、ターミナルにログインしたユーザをシミュレートします。つまり、各プロセス内のコマンドは、ターミナルからユーザが入力したコマンドと同じタイプのコマンドです。ロード・テストは連続して複数の独立プロセスを作成し、各独立プロセスがそれぞれのコマンド・プロセスを同時に実行します。つまり、独立プロセスと同じ数のユーザが同時にターミナルからコマンドを入力するのと同じような影響をシステムに与えます。このようにして、ロード・テストは、通常のシステムにおける使用と同じような環境を作り出します。

ロード・テストは、論理名 LOADS を使って、作成する独立プロセスの数を決定します。UETP コマンド・プロセスを起動すると、シミュレートするユーザの人数、つまり、作成する独立プロセスの数を尋ねてきます (5.5.3 項「ロード・テスト用のユーザ負荷の定義」を参照)。この数は、使用しているシステムのメモリ量、スワップ領域、およびページング領域を考慮して決める必要があります。このときのユーザの応答によって、グループ論理名 LOADS が定義されます。

UETP マスタ・コマンド・プロセスは、終了フェーズにおいて、テストによって行われたグループ論理名の割り当てを解除します。UETP パッケージが正常終了しなかった場合のみ、グループ論理名 LOADS の割り当ては解除されません。

作成する独立プロセスの数にもよりますが、ロード・テストによって実行されるコマンド・プロセスは大量の出力を生成します。各独立プロセス (すなわちユーザ) に対して、テストは UETLOnnnn.LOG と呼ばれる出力ファイルのバージョンを作成します (nnnn は数値文字列)。コンソールには、ロード・テストの進行を表す状態情報だけ表示されます。

ロード・テストが完全な UETP の一部として実行される場合も、独立したフェーズとして実行される場合も、UETP は UETLOnnnn.LOG ファイルを結合し、出力をファイル UETP.LOG に書き込み、最後に、個々の出力ファイルを削除します。

システム・ロード・テストを独立したフェーズとして実行するには、スタートアップ・ダイアログの中から LOAD を選択します (5.5.1 項「フェーズのサブセットの実行方法」を参照)。

5.9.4 DECnet for OpenVMS テスト・フェーズ

DECnet for OpenVMS ソフトウェアがユーザの OpenVMS システムに組み込まれている場合、完全な UETP の実行によって、DECnet ハードウェアおよびソフトウェアが自動的にテストされます。通信デバイスは DECnet に割り当てられ、DECnet デバイスは UETP デバイス・テストでテストできないので、DECnet for OpenVMS または他のアプリケーションがそのデバイスを割り当てている場合、UETP はイーサネット・アダプタをテストしません。DECnet テストの開始時、DECnet ノードおよびサーキット・カウンタはゼロにされ、実行時の障害監視が可能になります。

他の UETP フェーズと同様に、DECnet for OpenVMS フェーズを独立させて実行することもできます。5.5.1 項「フェーズのサブセットの実行方法」を参照してください。

5.9.4.1 環境

DECnet for OpenVMS テストは、DECnet がサポートするすべてのノード・タイプ (ルーティング・ノードおよび非ルーティング・ノードを含む) に接続された OpenVMS システム、および若干異なるタイプのオペレーティング・システム (RSTS, RSX, TOPS, RT など) 上で正常に動作します。遠隔システムには、システム間でファイルをコピーするための省略時のアクセス権が必要です。DECnet フェーズでは、次のテストを行います。

- UETP が実行されているノード。
- テストを行う遠隔ノードが論理名 UETP\$NODE_ADDRESS に定義されていない場合は、連続したすべてのサーキット。遠隔ノードが定義されている場合は、DECnet フェーズは 1 つのサーキットしかテストしない。
- すべての隣接ノードまたは第 1 ホップ・ノード、およびサーキット (同時)。

テストがサポートする通信回線の数には制限はありません。ある隣接ノードにおけるテストを、通常の通信転送率で 2 分以上継続してはなりません。



注意:

UETP は、ユーザのシステムが FAL オブジェクトに対して省略時のアクセス権を持っていることを想定しています。しかし、ネットワーク構成コマンド・プロシージャ NETCONFIG.COM は、省略時の設定で、FAL オブジェクトに対するアクセス権を提供しません。NETCONFIG.COM が提供する省略時の設定で DECnet ソフトウェアをインストールする場合、UETP DECnet フェーズでエラー・メッセージが表示されることがあります。このエラー・メッセージは無視して構いません。詳細は 5.8.12 項「FAL オブジェクトに対する省略時のアクセス権の欠如」を参照してください。

5.9.4.2 DECnet フェーズの動作

UETP (UETPHAS00.EXE の制御下において) は、ファイル UETDNET00.DAT を読み込み、DECnet for OpenVMS フェーズ中に次の手順を行います。

1. ネットワーク制御プログラム (NCP) の LOOP EXECUTOR コマンドを複数回実行し、UETP が動作しているノードをテストする。
2. NCP を使って、コマンド SHOW ACTIVE CIRCUITS を実行する。この結果は、UETININET.TMP に書き込まれ、そこから UETP はデータ・ファイル UETININET.DAT を作成する。UETININET.TMP ファイルには、ON 状態であるが遷移状態でないサーキットについての次の情報が入っている。
 - サーマット名
 - ノード・アドレス
 - ノード名 (存在する場合)

UETININET.TMP ファイルは、テストするデバイスを決断するために、DECnet フェーズ全体で使用される。

3. UETININET.TMP ファイルを使って、テスト可能なサーキットごとに NCP コマンド・プロシージャを 1 つ作成する。各コマンド・プロシージャには、サーキット・カウンタおよびノード・カウンタをゼロにし、ファイルのコピーによってサーキットおよび隣接ノードをテストする複数の NCP コマンドが入っている。



注意:

カウンタをゼロにしたくない場合は、DECnet for OpenVMS ソフトウェアをテストしないでください。

4. 手順 3 のコマンド・プロシージャを並行して実行し、ユーザ負荷が高い状態をシミュレートする。シミュレートされるユーザ負荷は、以下の値のうち少ない方である。
 - テスト可能なサーキットの数に 2 をかけたもの
 - 資源が不足するまでにシステムに作成できるユーザ独立プロセスの最大数 (UETINIT00 で決定する)
5. プログラム UETNETS00.EXE を実行する。このプログラムは UETININET.DAT ファイルを使って、テスト可能な各サーキットに対するサーキット・カウンタおよびノード・カウンタをチェックする。カウンタが劣化を示している場合 (つまり、ゼロではない場合)、その名前と値がコンソールに報告される。ログ・ファイルには、すべてのカウンタが報告されるが、劣化を示すカウンタだけはコンソールにも報告される。次に、UETNETS00 出力の例を示す。

```
%UETP-S-BEGIN, UETNETS00 beginning at 22-JUN-2000 13:45:33.18
%UETP-W-TEXT, Circuit DMC-0 to (NODENAME1) OK.
%UETP-I-TEXT, Node (NODENAME2) over DMC-1 response timeouts = 1.
%UETP-I-TEXT, Circuit DMC-1 to (NODENAME2) local buffer errors = 34.
%UETP-I-TEXT, Node (NODENAME3) over DMP-0 response timeouts = 3.
%UETP-S-ENDED, UETNETS00 ended at 22-JUN-2000 13:45:36.34
```

カウンタの劣化が必ずしもエラーの原因となるわけではないので、テストが成功したかどうかは、システムではなく、ユーザが判断することである。次のカウンタは劣化を示す。

サーキットの場合

- Arriving congestion loss
- Corruption loss
- Transit congestion loss
- Line down
- Initialization failure
- Data errors inbound
- Data errors outbound
- Remote reply timeouts
- Local reply timeouts
- Remote buffer errors
- Local buffer errors
- Selection timeouts
- Remote process errors
- Local process errors
- Locally initiated resets
- Network initiated resets

ノードの場合

- Response timeouts
- Received connect resource errors
- Node unreachable packet loss
- Node out of range packet loss
- Oversized packet loss
- Packet format error
- Partial routing update loss
- Verification reject

5.9.5 クラスタ統合テスト・フェーズ

クラスタ統合テスト・フェーズは、DECnet for OpenVMS ソフトウェアに大きく依存する 1 つのプログラムおよび 1 つのコマンド・ファイルから構成されます。このフェーズは、DECnet for OpenVMS ソフトウェアを使って、クラスタ中の各 OpenVMS ノード上に SYSTEST_CLIG プロセスを作成し、各ノードと通信します。SYSTEST_CLIG は SYSTEST に類似するアカウントです。しかし、SYSTEST_CLIG はクラスタ統合テストでしか使用されません。クラスタ・テスト・フェーズを正しく実行するには、次の SYSTEST_CLIG アカウントの制限が必要です。

- アカウントが使用可能であり、パスワードがヌルでなければならない。詳細は 5.3.15 項「OpenVMS Cluster のテスト」を参照。
- SYSTEST アカウントと同じ UIC でなければならない。
- SYSTEST アカウントと同じ特権およびクォータでなければならない。詳細は 5.8.2 項「クォータ、特権、アカウントの間違い」を参照。
- DECnet for OpenVMS ソフトウェアからのみログインできるアカウントでなければならない。
- ログイン時、実行中の UETCLIG00.COM にロックされるアカウントでなければならない。

これらの項目は、システムの機密保護およびプライバシーを守るために必要です。SYSTEST_CLIG プロセスが OpenVMS ノード上に作成できない場合は、障害の原因が示され、ファイル・テスト中にはロック・テストおよび 共用アクセス用のノードが無視されます。また、このテスト

では、SYSTEST_CLIG プロセスが作成されなかったノードからのログ・ファイルのコピーは行われません。SYSTEST_CLIG プロセスの作成後、SYSTEST_CLIG プロセスとの通信に問題が発生した場合、それ以降のロック・テストおよびファイル共有テストでは、このノードは除外されます。クラスタ統合テストの終了時、当該ノードでエラーが発生したかどうか報告されません。

UETCLIG00.EXE は、1 次スレッドと 2 次スレッドという、2 つの実行用スレッドを持っています。1 次スレッドは、クラスタ構成 (OpenVMS ノード、HSC ノード、およびテストを実行するノードで使用できる取り付けディスク) をチェックします。選択された OpenVMS ノードの場合、1 次スレッドは DECnet ソフトウェアを使って SYSTEST_CLIG プロセスを起動しようとしています。1 次スレッドが SYSTEST_CLIG プロセスをノード上で起動できた場合、そのノードはコマンド・ファイル UETCLIG00.COM を起動します。このコマンド・ファイルは、UETCLIG00.EXE を起動し、2 次実行スレッドを実行します。

1 次スレッドを実行しているプロセスは、2 次スレッドを実行しているプロセスと通信できるかどうかチェックします。次に、ロックを外し、デッドロック状態が作成されるよう、2 次スレッドに命令します。

1 次スレッドは、クラスタ中で選択した OpenVMS および HSC ノード上の同じディスクにファイルを作成しようとしています。1 次スレッドはブロックを書き込み、そのブロックを再び読み取り、最後に、そのブロックを確認します。次に、OpenVMS ノードを 1 つランダムに選択し、そのブロックの読み込みと確認を行うよう指示します。次に、1 次スレッドは他のブロックを書き込むことによって、ファイルを拡大し、2 次ブロックは 2 番目のブロックを読み込み、確認します。このファイルは削除されます。

2 次プロセスは終了します。2 次プロセスは、自身の SYS\$ERROR ファイルの内容を 1 次プロセスにコピーするので、UETP ログ・ファイルおよびコンソール・レポートはすべての問題を 1 つの場所に表示することができます。DECnet for OpenVMS ソフトウェアは、テストの実行時に自動的に SYS\$TEST の中に NETSERVER.LOG を作成します。したがって、必要であれば、後でこのノードのファイルを読むことができます。

テストの実行中、1 次プロセスはシステム・サービス SYS\$BRKTHRU を使って、テストの開始と終了を各 OpenVMS ノードのコンソール・ターミナルに知らせます。

グループ論理名 MODE を文字列 DUMP と同等に定義することによって、ほとんどのイベントの発生を追跡することができます。論理名の定義は、定義されたノード上でしか適用されません。MODE は、イベントを追跡したいクラスタ中の各ノード上で定義しなければなりません。

第6章 システムに関する情報の入手

この章では、システム・ログ・ファイルの設定と管理、エラー・ログ・ファイルの管理、およびシステム管理ユーティリティを使ったシステムの監視について説明します。

この章では、次の作業を説明します。

作業	参照箇所
エラー・フォーマット (ERRFMT) によるエラー・ログ・ファイルの作成方法	6.3 項 「エラー・ログ・フォーマットの 使用法」
Error Log Viewer (ELV) によるレポートの作成方法	6.4 項 「Error Log Viewer (ELV) の使用 方法」
オペレータ・ログ・ファイルの設定、管理、プリント	6.5 項 「オペレータ・ログ・ファイルの 設定、管理、プリント」
機密保護監査機能の使用法	6.6 項 「機密保護監査機構の使用法」
MONITOR ユーティリティを使用したシステムの性能の監視	6.7 項 「オペレーティング・システムの 性能の監視」

さらに、次の項目について説明します。

項目	参照箇所
システム・ログ・ファイル	6.1 項 「システム・ログ・ファイルについて」
エラー・ログ機構	6.2 項 「エラー・ログ機構」
Error Log Viewer (ELV) ユーティリティ	6.4.1 項 「Error Log Viewer (ELV) について」
オペレータ・ログ・ファイル	6.5.1 項 「オペレータ・ログ・ファイルについて」
OPCOM メッセージ	6.5.2 項 「OPCOM メッセージについて」
機密保護監査機構	6.6.1 項 「機密保護監査機構について」
Monitor ユーティリティ	6.7.1 項 「MONITOR について」

6.1 システム・ログ・ファイルについて

システム管理を行う場合、システム・イベントに関する情報を収集して検討することが必要になります。OpenVMS オペレーティング・システムでは、システム資源の使用状況、エラー状態、他のシステム・イベントの情報を記録するログ・ファイルがいくつか提供されます。これらのログ・ファイルについて表 6-1 「システム・ログ・ファイル」で簡単にまとめます。

表 6-1 システム・ログ・ファイル

ログ・ファイル	説明	参照箇所
エラー・ログ・ファイル	このファイルには、デバイスおよび CPU に関するエラー・メッセージが自動的に記録される。	6.2 項 「エラー・ログ機構」
オペレータ・ログ・ファイル	このファイルには、オペレータ通信マネージャ (OPCOM) によって、システム・イベントが記録される。	『OpenVMS システム管理者マニュアル (上巻)』、 6.5 項 「オペレータ・ログ・ファイルの設定、管理、プリント」
会計情報ファイル	このファイルには、システム資源の使用状況が記録される。	第7章 「資源使用状況の調査」
機密保護監査ログ・ファイル	監査サーバ・プロセスにより、機密保護関係のシステム・イベントが書き込まれる。	6.6 項 「機密保護監査機構の使用法」

6.2 エラー・ログ機構

エラー・ログ・サブシステムは、自動的にエラー・メッセージを最新バージョンのエラー・ログ・ファイル `SYS$ERRORLOG:ERRLOG.SYS` に書き込みます。エラー・ログ・レポートは、主に、弊社のサポート担当者がハードウェアの問題箇所を特定するときに使用するものです。また、システム管理者でも、あるシステム障害が頻繁に発生する場合には、それが注意を要するものであるかどうか、エラー・ログ・レポートから判断することができます。

エラー・ログ・サブシステムについて

エラー・ログ・サブシステムは、表 6-2 「エラー・ログ・サブシステムの構成要素」に示す 3 つの部分から構成されます。

表 6-2 エラー・ログ・サブシステムの構成要素

構成要素	説明
エグゼクティブ・ルーチン	エラーおよびイベントを検出し、関連する情報をメモリ内のエラー・ログ・バッファに書き込む。
エラー・フォーマッタ (ERRFMT)	ERRFMT プロセス は、システムのブート時に起動し、定期的エラー・ログ・バッファを空にするとともに、エラーの記述を標準形式に変形し、書式化した情報をシステム・ディスク上のエラー・ログ・ファイルに格納する (6.3.2 項「エラー・ログ・ファイルの管理」を参照)。 エラー・フォーマッタを使えば、ERRFMT プロセスが回復不可能なエラーに遭遇してプロセス自身を削除する場合に、SYSTEM アカウントや他のユーザにメールを送ることができる (6.3.3 項「ERRFMT によるメールの送信」を参照)。
Error Log Viewer (ELV)	エラー・ログ・ファイルの内容を選択して報告する。このユーティリティは、バージョン 7.3 およびそれ以降の OpenVMS を実行しているシステム上で書き込まれたエラー・ログの場合に最も便利である。 ELV を呼び出すには、DCL の <code>ANALYZE/ERROR_LOG/ELV</code> コマンドを入力する (6.4 項「Error Log Viewer (ELV) の使用方法」を参照)。

エグゼクティブ・ルーチンとエラー・フォーマッタ (ERRFMT) プロセスは継続的に稼働します。途中、ユーザと対話することはありません。エグゼクティブ・ルーチンは、エラーやイベントを検出すると、そのままの形でメモリ上のエラー・ログ・バッファに格納します。エラー・ログ・バッファの 1 つがいっぱいになるか、またはあらかじめ設定された時間が経過すると、ERRFMT は自動的にエラー・ログ・バッファを `SYS$ERRORLOG:ERRLOG.SYS` に書き込みます。しかし、エラーが瞬時に大量に発生し、ERRFMT プロセスがエラー・ログ・バッファを空にする前に、エラー・ログ・バッファがあふれてしまうこともあります。この状態を発見するためには、エラー・ログ・レポートを読んで、レコード番号の途中で抜けた箇所がないかどうか調べます。ERRFMT プロセスがエラー・ログ・バッファ空間を解放すると、ただちに、エラー・ログは再開されます。

エラーが多すぎてエラー・ログ・ファイルに書き込めなくなると、ERRFMT プロセスは、システム・コンソール・ターミナルにエラー・メッセージを表示し、自分で実行を停止します。ERRFMT プロセスを再開する方法については、6.3.1 項「ERRFMT プロセスの再起動」を参照してください。

6.3 エラー・ログ・フォーマッタの使用法

エラー・ログ・フォーマッタ (ERRFMT) プロセスは、ブート時に自動的に起動されます。次に各作業の実行方法と参照箇所を示します。

作業項目	参照箇所
ERRFMT プロセスの再起動 (必要な場合)	6.3.1 項 「ERRFMT プロセスの再起動」
エラー・ログ・ファイルの管理	6.3.2 項 「エラー・ログ・ファイルの管理」
ERRFMT プロセスが削除されたときのメールの送信	6.3.3 項 「ERRFMT によるメールの送信」

6.3.1 ERRFMT プロセスの再起動

ERRFMT プロセスを再起動するためには、次の手順に従ってください。

1. システム管理者のアカウントでログインする。これは、操作に必要な特権を獲得するためである。
2. コマンド・パラメータとして ERRFMT を指定し、サイト別スタートアップ・コマンド・プロシージャ (STARTUP.COM) を実行する。

```
$ @SYS$SYSTEM:STARTUP ERRFMT
```



注意:

システム・ディスク上でディスク・クォータが使用可能になっている状態で、UIC [1,4] が十分なクォータを持っている場合にのみ、ERRFMT は起動されます。

6.3.2 エラー・ログ・ファイルの管理

エラー・ログ・ファイル SYS\$ERRORLOG:ERRLOG.SYS は共用ファイルのため、ANALYZE/ERROR_LOG ユーティリティがそのファイルの既存のエラー・ログ・エントリを読み込んだり抽出したりしている間にも、ERRFMT プロセスは新しいエントリを書き込みます。

ERRLOG.SYS は、システム管理者が明示的にリネームするか削除するまで大きくなっていきます。したがって、エラー・ログ・ファイルは定期的に整理する必要があります。1つの方法として、1日1回 ERRLOG.SYS をリネームします。そうすると、新しいエラー・ログ・ファイルが自動的に作成されます。たとえば、毎朝9時に ERRLOG.SYS を ERROLOG.OLD にリネームします。リネームした古いエラー・ログ・ファイルは別のボリュームに移すか、システム・ディスクから削除すれば、システム・ディスクの空間が解放されます。

また、論理名 SYS\$ERRORLOG をエラー・ログ・ファイルを格納したいデバイスとディレクトリに定義することにより、システム・ディスク以外のディスク上にエラー・ログ・ファイルを格納する方法もあります。次に例を示します。

```
$ DEFINE/SYSTEM/EXECUTIVE SYS$ERRORLOG DUA2:[ERRORLOG]
```

システムを起動するたびにこの論理名を定義する場合は、論理名定義を SYLOGICALS.COM プロシージャに追加します。詳細は『OpenVMS システム管理者マニュアル (上巻)』を参照してください。

このとき、誤って使用中のエラー・ログ・ファイルを削除しないように注意してください。また、リネームするときに、ファイル名の先頭あるいは最後に日付を付けて整理する方法もあります。

6.3.3 ERRFMT によるメールの送信

エラー・フォーマッタ (ERRFMT) を使用すれば、ERRFMT プロセスが回復不可能なエラーを検出し、プロセス自身を削除するときに、システム管理者や他の指定したユーザにメールを送ることができます。

2つのシステム論理名 ERRFMT\$_SEND_MAIL および ERRFMT\$_SEND_TO が、この機能を制御します。

- ERRFMT\$_SEND_MAIL
メールの送信を可能にするには、文字列 TRUE (大文字と小文字の区別なし) を指定する。他の値では、メールの送信はできない。
- ERRFMT\$_SEND_TO
ユーザ名を指定する (現在の省略時の値は SYSTEM)。
配布リストおよび複数のユーザ名はなるべく使用しないこと。

上記の論理名を定義するには、次のいずれかを行います。

- DCL の DEFINE/SYSTEM コマンドを使用し、動的に定義する。
変更後、ERRFMT を停止し再起動しなければ、変更は有効にならない。
- SYS\$STARTUP:SYLOGICAL.COM を使用し、恒久的に定義する。
定義した論理名は、次回システムをリブートしたときから有効となる。これ以降の説明では、この方法を使用する。

6.3.3.1 メールを送信するための ERRFMT の停止と再起動

ERRFMT\$_SEND_MAIL が TRUE と定義されている場合、ユーザは、ERRFMT がプロセス自身を削除するという表題のメール・メッセージを受信します。オペレータ・ログ・ファイルおよびシステム・コンソール OPA0: には、発生した障害についての詳細な情報、および ERRFMT を再起動するための指示が出力されます。しかし、ユーザはコンソールにいないこともあるので、この情報を見ないかもしれません。

たとえば、メールの送信が可能なモードで ERRFMT を使用している場合、メールの送信を禁止にするには、システム管理者のアカウントを使って、SYS\$STARTUP:SYLOGICAL.COM を編集し、次のコマンドを追加します。

```
$ DEFINE/SYSTEM ERRFMT$_SEND_MAIL FALSE
```

もう一度、メールの送信を可能にするには、システム管理者のアカウントを使って、SYS\$STARTUP:SYLOGICAL.COM を編集し、次のコマンドを追加します。

```
$ DEFINE/SYSTEM ERRFMT$_SEND_MAIL TRUE
```

6.3.3.2 メールを他のユーザに送信する方法

省略時の設定では、メールは SYSTEM アカウントに送信されます。しかし、ERRFMT\$_SEND_TO を定義すれば、ERRFMT が自分自身を削除するときに、メールを他のユーザに送信することもできます。

メールを受信するユーザ名を変更するには、システム管理者のアカウントを使って、SYS\$STARTUP:SYLOGICAL.COM を編集し、適切な論理名を定義する DEFINE コマンドを追加します。次に例を示します。

```
$ DEFINE/SYSTEM ERRFMT$_SEND_TO R_SMITH
```

配布リストおよび複数ユーザ名はなるべく使用しないでください。

6.4 Error Log Viewer (ELV) の使用方法

Error Log Viewer (ELV) は、エラー・ログ・ファイルの内容を選択して報告します。ELV は、OpenVMS バージョン 7.3 およびそれ以降を実行しているシステム上で書き込まれたエラー・ログで使用すると最も便利です。

ELV についての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』の ELV についての章を参照してください。

6.4.1 Error Log Viewer (ELV) について

Error Log Viewer (ELV) ユーティリティを使用すると、エラー・ログ・ファイルを、ユーザが読める形式で、コマンド行から素早く調べることができます。この処理を実行することで、データに対し、System Event Analyzer (SEA) などのツールによるより分かりやすい分析が必要かどうかを判断できます。

ELV は、表 6-3 「ELV が完全にサポートするイベント・タイプ」に示すイベント・タイプに属するすべてのイベントについて、詳細情報を出力します。イベント・タイプは、同じ表に示すイベント・クラスでグループ分けされます。

表 6-3 ELV が完全にサポートするイベント・タイプ

イベント・クラス	イベント・タイプ
制御エントリ	システム起動、タイム・スタンプ、オペレータおよびネットワークのメッセージ、通知イベント、ERRLOG.SYS が作成したメッセージ、および Send Message から Error Logger (\$SENDERR) システム・サービスへのメッセージ
ボリューム変更	ボリュームのマウントとディスマウント
バグチェック	システムのバグチェック、ユーザのバグチェック、およびクラッシュ再起動
マシン・チェック	訂正可能エラーの絞り込み通知、6A0/6B0 回復可能な訂正不能エラー
デバイス・エラー	ソフトウェア・パラメータ

ELV は、表 6-4 「ELV が部分的にサポートするイベント・タイプ」に示すイベント・タイプに属する一部のイベントについて、詳細情報を出力します。イベント・タイプは、同じ表に示すイベント・クラスでグループ分けされます。

表 6-4 ELV が部分的にサポートするイベント・タイプ

イベント・クラス	イベント・タイプ
マシン・チェック	620 システムの訂正可能エラー、630 プロセッサの訂正可能エラー、660 システムの訂正不能エラー、670 プロセッサの訂正不能エラー、680 システム・イベント、コンソール・データ・ログ
デバイス・エラー	デバイス・エラー、デバイス・タイムアウト、非同期デバイス・アテンション
非要求型 MSCP	ログに記録される MSCP メッセージ

6.4.2 ELV の起動

ELV を起動するには、次の DCL コマンドを入力します。

```
$ ANALYZE/ERROR_LOG/ELV
```

ELV コマンドを入力しないと、ユーティリティは会話型シェル・モードに入り、ELV プロンプトを表示します。

```
ELV>
```

この後、ELV コマンドを入力できます。ELV はコマンドを実行した後、再度 ELV> プロンプトを表示します。

ELV プロンプトから ELV コマンドを実行した後に DCL に直接戻るには、/NOINTERACTIVE 修飾子を使用します。

また、ELV コマンドを DCL から直接入力することもできます。例を次に示します。

```
$ ANALYZE/ERROR_LOG/ELV TRANSLATE ERRLOG.SYS;42
```

特に指定しなければ、ELV はコマンドを実行した後、DCL プロンプトに戻ります。

DCL から直接 ELV コマンドを実行した後に会話型シェル・モードに入るには、/INTERACTIVE 修飾子を使用します。

6.4.3 主な ELV コマンド

表 6-5 「主な ELV コマンド」に、主な ELV 操作のコマンドを示します。

表 6-5 主な ELV コマンド

コマンド	説明
CONVERT	新しい形式で書き込まれている 1 個以上のバイナリ・エラー・ログ・ファイルのイベントを、古い形式に変換して 1 個の新しいエラー・ログ・ファイルに書き込む。新しいファイルは、ANALYZE/ERROR_LOG により読み込めるようになる。 このコマンドは、ELV が変換をサポートしていない古いエラー・ログ・イベントの変換を可能とするために主に使用される。
DUMP	1 個以上のバイナリ・エラー・ログ・ファイルのイベントを、OpenVMS ダンプ・スタイル形式で 1 個の新しい ASCII 出力ファイルに書き込む。
TRANSLATE	1 個以上のバイナリ・エラー・ログ・ファイルのイベントに対してバイナリからテキストへの変換を行い、結果として得られたレポートを、ターミナルまたは 1 個の新しい ASCII 出力ファイルに書き込む。
WRITE	1 個以上のバイナリ・エラー・ログ・ファイルのイベントを、1 個の新しいバイナリ・エラー・ログ・ファイルにコピーする。

これらのすべてのコマンドに共通の修飾子を使用して、コマンドで処理されるイベントを選択または拒否することができます。たとえば、TRANSLATE /SINCE=YESTERDAY を指定すると、昨日以降に発生した有効なイベントすべてが変換されます。

6.4.4 TRANSLATE コマンドを使用した標準レポート

TRANSLATE コマンドを使用すると、さまざまな詳細レベルで標準レポートを作成できます。標準レポートの作成は、ELV ユーティリティの主要な機能です。

標準レポートの詳細レベルを指定するには、TRANSLATE コマンドに /BRIEF、/FULL、または /ONE_LINE 修飾子を指定します (概要は表 6-6 「標準レポートの詳細レベル」を参照)。または、詳細レベル修飾子を省略して、省略時のレポートを表示します。これらの修飾子の他に、/TERSE 修飾子を使用すると、詳細レベルに関係なく、データの加工が少ない標準レポートを得ることができます。

表 6-6 標準レポートの詳細レベル

詳細レベル	修飾子	説明
1 行	/ONE_LINE	ヘッダ情報だけを、標準レポートに含める。
簡潔	/BRIEF	ヘッダ情報の他に、最も重要な情報だけを含める。
省略時の表示	(なし)	ヘッダ情報の他に、一般的に役立つイベント情報だけを含める。
完全	/FULL	ヘッダ情報の他に、すべてのイベント情報を含める。

6.4.4.1 標準レポートの例

例 6-1 「標準レポートと要約レポート」に、標準レポートと要約レポートの例を、順に示します。標準レポートだけを作成する (省略時の指定で含まれる要約レポートを作成しない) には、/NOSUMMARY 修飾子を使用します。要約レポートだけを作成する (省略時の指定で含まれる標準レポートを作成しない) には、/SUMMARY 修飾子を使用します。

例 6-1 標準レポートと要約レポート

Output for SYS\$COMMON:[SYSEXE.ERRLOGS]EXAMPLE.DAT;1

EVENT	EVENT_TYPE	TIMESTAMP	NODE	EVENT_CLASS
1	Volume Mount	14-AUG-2003 13:31:39.12	FRANZ	VOLUME_CHANGES

DESCRIPTION	RANGE	VALUE	TRANSLATED_VALUE
Hardware Architecture		4	Alpha
Hardware System Type		35	Wildfire
Logging CPU		3	
Number of CPU's in Active Set		4	
System Marketing Model		1968	COMPAQ AlphaServer GS160
Error Mask	<31:006gt;;	0x00000003	
Seconds Since Boot		17	
Error Sequence Number		46	
DSR String		AlphaServer GS160 6/731	
Operating System Version		X9WY-SSB	
Owner UIC of the Volume		65537	
Unit Operation Count		378	
Device Unit Number		200	
Device Generic Name		FRANZ\$DKB	
Volume Number within Set		0	
Number of Volumes within Set		0	
Volume Label		OPAL_X9WY	

ERROR_LOG_SUMMARY	
Total number of events:	1
Number of the first event:	1
Number of the last event:	1
Earliest event occurred:	14-AUG-2003 13:31:39.12
Latest event occurred:	14-AUG-2003 13:31:39.12
Number of events by event class:	
VOLUME_CHANGES	1

6.5 オペレータ・ログ・ファイルの設定、管理、プリント

この節では、オペレータ・ログ・ファイルと、このファイルに記録される OPCOM メッセージについて説明し、次の表の作業手順を示します。これらの作業を行う場合は OPER 特権が必要になります。

作業	参照箇所
オペレータ・ログ・ファイルの設定	6.5.3 項 「オペレータ・ログ・ファイルの設定」
オペレータ・ログ・ファイルの管理	6.5.4 項 「オペレータ・ログ・ファイルの管理」
オペレータ・ログ・ファイルのプリント	6.5.5 項 「オペレータ・ログ・ファイルのプリント」

6.5.1 オペレータ・ログ・ファイルについて

オペレータ・ログ・ファイル (SYS\$MANAGER:OPERATOR.LOG) には、システム・イベントと、オペレータ通信マネージャ (OPCOM) からオペレータ・ターミナルに送信されたユーザ要求が記録されます。記録は、すべてのオペレータ・ターミナルが使用不能になっている場合でも行われます。通常、OPCOM はシステムをブートすると起動します。OPCOM についての詳細は、『OpenVMS システム管理者マニュアル (上巻)』を参照してください。

オペレータ・ログ・ファイルにより、ハードウェアおよびソフトウェアの障害を予測してそれらを事前に防止したり、ディスクおよび磁気テープに対するユーザ要求を監視することもできます。オペレータ・ログ・ファイルを定期的に調べれば、今後障害につながる可能性がある問題を事前に発見し、適切な処置をとることができます。

OPERATOR.LOG ファイル (または論理名 OPC\$LOGFILE_NAME が指すファイル) のサイズとアクセスは、それが置かれているディスク・デバイスのサイズとアクセスの制限を受けます。ディスク・デバイスにログ・ファイルを書き込むだけの余裕がなかったり、他の方法でのデバイスへのアクセスが制限されていたりすると、ログ・ファイルから記録が失われる可能性があります。

6.5.2 OPCOM メッセージについて

次の表に示す各項では、オペレータ・ログ・ファイルに含まれるメッセージの種類を説明します。

メッセージの種類	参照箇所
初期化メッセージ	6.5.2.1 項「初期化メッセージ」
デバイス状態を示すメッセージ	6.5.2.2 項「デバイス状態メッセージ」
ターミナルの使用可能または不能メッセージになったことを示すメッセージ	6.5.2.3 項「ターミナルの使用可能または不能メッセージ」
ユーザ要求とオペレータ応答メッセージ	6.5.2.4 項「ユーザ要求とオペレータ応答メッセージ」
ボリュームがマウントまたはディスマウントされたことを示すメッセージ	6.5.2.5 項「ボリュームがマウントまたはディスマウントされたことを示すメッセージ」
システム・パラメータ・メッセージ	6.5.2.6 項「システム・パラメータ・メッセージ」
機密保護アラーム・メッセージ	6.5.2.7 項「機密保護アラーム・メッセージ」

6.5.2.8 項「オペレータ・ログ・ファイルの内容」に、オペレータ・ログ・ファイル内にある典型的なメッセージの例を示します。

6.5.2.1 初期化メッセージ

REPLY/LOG コマンドを入力すると、現在のオペレータ・ログ・ファイルがクローズし、代わりに新しいバージョンがオープンします。それ以後生成される OPCOM メッセージは、この新しいログ・ファイルに記録されるようになります。

新しいログ・ファイルを作成すると、その先頭には初期化メッセージが記録されます。初期化メッセージには、ログ・ファイルを初期化したオペレータとそのログ・ファイル指定が次の形式で示されます。

```
%%%%%%%%%% %OPCOM, <dd-mmm-yyyy hh:mm:ss.cc> %%%%%%%%%%%  
Logfile has been initialized by operator <terminal-name>  
Logfile is <logfile-specification>
```

例

```
%%%%%%%%%% OPCOM, 19-APR-2002 12:29:24.52 %%%%%%%%%%%  
Logfile has been initialized by operator _MARS$VTA2:  
Logfile is HOMER::SYS$SYSMOND:[SYSMGT] OPERATOR.LOG;43
```

6.5.2.2 デバイス状態メッセージ

一部の入出力ドライバは、制御するデバイスの状態変化に関するメッセージを OPCOM に送信します。たとえば、ライン・プリンタがオフラインになった場合、明示的にオンライン状態に戻すまで、オペレータ・ログ・ファイルには OPCOM メッセージが定期的に記録されます。オペレータ・ログ・ファイルに記録されるデバイス状態メッセージの形式は次のとおりです。

```
%%%%%%%%%% OPCOM <dd-mmm-yyyy hh:mm:ss.cc> %%%%%%%%%%%  
Device <デバイス名> is offline
```

このメッセージは、カード・リーダー、ライン・プリンタ、磁気テープの場合に表示されます。

6.5.2.3 ターミナルの使用可能または不能メッセージ

次に、ターミナルをオペレータ・ターミナル(コンソール)として使用可能または使用不能にするコマンドの例を示し、それらのコマンドを実行したときにオペレータ・ログ・ファイルに記録されるメッセージについて説明します。

REPLY/ENABLE メッセージ

ターミナルをオペレータ・ターミナルとして指定する場合は、使用したいターミナルで REPLY/ENABLE と入力します。OPCOM は次の形式のメッセージをオペレータ・ターミナルに表示し、同時にオペレータ・ログ・ファイルに記録して、この要求を確認します。

```
%%%%%%%%%%%% %OPCOM dd-mmm-yyyy hh:mm:ss.cc %%%%%%%%%%%%%%
Operator <ターミナル名> has been enabled, username <ユーザ名>
```

```
%%%%%%%%%%%% %OPCOM dd-mmm-yyyy hh:mm:ss.cc %%%%%%%%%%%%%%
Operator status for operator <ターミナル名>
<状態レポート>
```

このメッセージはオペレータ・ターミナルとして使用可能になったターミナルを示し、そのターミナルが受け付けて応答できる要求をリストします。

また、REPLY/ENABLE= クラス・コマンドを入力すると、ターミナルを特定の機能のためのオペレータ・ターミナルとして指定することもできます。

たとえば、REPLY/ENABLE=TAPES コマンドを入力すると、OPCOM により、次のようなメッセージが表示されます。

```
%%%%%%%%%%%% %OPCOM 19-APR-2002 10:25:35.74 %%%%%%%%%%%%%%
Operator _ROUND$OPA1: has been enabled, username SYSTEM
```

```
%%%%%%%%%%%% %OPCOM 19-APR-2002 10:25:38.82 %%%%%%%%%%%%%%
Operator status for operator _ROUND$OPA1:
TAPES
```

OPCOM は、このターミナルがオペレータ・ターミナルとして使用可能になったことを確認し、またこのターミナルがテープのマウントやディスマウントなど、磁気テープに関するイベントの要求と応答だけを扱うことのできるターミナルであることを示します。

REPLY/DISABLE メッセージ

オペレータ・ターミナルとして宣言されたターミナルは、そのオペレータがログアウトすると自動的に非オペレータ・ターミナルの状態になります。ログアウトしないでターミナルを通常の(非オペレータの)状態に戻すためには、そのターミナルから REPLY/DISABLE コマンドを実行します。

OPCOM は、そのターミナルがオペレータ・ターミナルでなくなったことを確認するメッセージをターミナルに表示し、同時にオペレータ・ログ・ファイルに記録します。このメッセージは、ターミナルが非オペレータ状態に戻ったことと、その状態変化が行われた日時を次の形式で示します。

```
%%%%%%%%%%%% %OPCOM <dd-mmm-yyyy hh:mm:ss.cc> %%%%%%%%%%%%%%
Operator <ターミナル名> has been disabled, username <ユーザ名>
```

あるターミナルをオペレータ・ターミナルとして宣言したときにその機能の一部が使用できない場合は、OPCOM から状態メッセージが表示されます。状態メッセージは、該当するターミナルが受け取ったり応答したりできる要求を示します。形式は次のとおりです。

```
%%%%%%%%%%%% %OPCOM <dd-mmm-yyyy hh:mm:ss.cc> %%%%%%%%%%%%%%
Operator status for operator <ターミナル名>
<状態レポート>
```

たとえば、ターミナルを、磁気テープおよびディスクに関連するメッセージ、および OPER10 というシステム固有の特殊なオペレータ・クラスのメッセージを受信するオペレータ・ターミ

ナルとして定義し、後になって、テープに関連するメッセージの受信をやめる場合を考えてみます。REPLY/DISABLE=TAPES コマンドを実行すると、OPCOM から次のようなメッセージが返されます。

```
%%%%%%%%%% %Opcom 19-APR-2002 09:23:45.32 %%%%%%%%%%
Operator status for operator TTA3
DISKS, OPER10
```

このメッセージは、TTA3 というターミナルが、ディスクに関するメッセージとオペレータ・クラス OPER10 へのメッセージを受信していること、および受信したメッセージに対して応答ができることを示しています。

6.5.2.4 ユーザ要求とオペレータ応答メッセージ

ユーザがオペレータと連絡をとるためには、REQUEST コマンドで /REPLY 修飾子あるいは /TO 修飾子のいずれかを指定します。これらの修飾子には次の働きがあります。

コマンド	説明
REQUEST/REPLY	<p>要求は次の形式でオペレータ・ログ・ファイルに記録される。</p> <pre>%%%%%%%%%% %OPCOM <dd-mmm-yyyy hh:mm:ss.cc> %%%%%%%%%% Request <要求 id>, from user <ユーザ名> on <ノード名><_ターミナル名:>, <"メッ セージ・テキスト"></pre> <p>このメッセージは、メッセージを送信したユーザ、メッセージが送信された時刻、メッセージに割り当てられた要求識別番号(要求ID)、そのメッセージの発信元のターミナル、およびメッセージを示している。</p>
REQUEST/TO	<p>要求内容は、REQUEST/REPLY の例で示した形式でオペレータ・ログ・ファイルに記録される。ただし、要求 ID は記録されない。</p> <pre>%%%%%%%%%% %OPCOM, <dd-mmm-yyyy hh:mm:ss.cc> %%%%%%%%%% Request from user <ユーザ名> on <ノード名><_ターミナル名:>, <"メッセージ・テ キスト"></pre>

また、メッセージは、ユーザに対してどのように応答するかによっても異なってきます。

コマンド	説明
REPLY/TO	<p>内容はオペレータ・ログ・ファイルに次の形式で記録される。</p> <pre>response message <hh:mm:ss.cc>, request <要求 id> completed by operator <ターミナル名></pre> <p>このメッセージは、オペレータがユーザの要求にどう応答したか、応答が入力された時刻、応答したオペレータを示している。</p>
REPLY/ABORT	<p>内容はオペレータ・ログ・ファイルに次の形式で記録される。</p> <pre><hh:mm:ss.cc>, request <要求 id> was aborted by operator <ターミナル名></pre>
REPLY/PENDING	<p>内容はオペレータ・ログ・ファイルに記録されない。これは、その要求が完了しない(すなわち、応答も中断もされない)ためである。</p>

REQUEST/REPLY コマンドによってすべてのオペレータ・ターミナルが使用不能になると、OPCOM はそれ以後、ユーザからのすべての要求をログ・ファイルに記録しますが、オペレータの応答が行われないことを示すメッセージをユーザに返すことはありません。

オペレータ・ログ・ファイルに記録されるオペレータ応答は、REPLY/ENABLE、REPLY/DISABLE、REPLY/LOG の各コマンドに関連するものだけです。他のオペレータ応答は記録されません。

6.5.2.5 ボリュームがマウントまたはディスマウントされたことを示すメッセージ

オペレータ・ログ・ファイルに記録されるオペレータ・メッセージの多くは、おそらく次のようなボリュームのマウントとディスマウントに関するものです。

```
%%%%%%%%%% OPCOM, 19-APR-2002 22:41:07.54 %%%%%%%%%%%
message from user SYSTEM
Volume "KLATU" "dismounted, on physical device MTA0:
15-APR-2002 22:42:14.81, request 2 completed by operator OPA0
```

6.5.2.6 システム・パラメータ・メッセージ

ユーザは適切な特権を持っていれば、システム・パラメータの以下の値を変更することができます。

値	説明
現在値	ディスク上の省略時のパラメータ・ファイルに格納されている値で、システムのブート時に使用される。
アクティブ値	メモリに格納されている値で、システムの稼働時に使用される。

システムはブートするとき、現在値をメモリに読み込み、アクティブ値を作成します。アクティブ値と現在値は、どちらかを変更するまでは同じ値となります。

ユーザは、システム・パラメータのアクティブ値と現在値に以下の変更を加えることができます。

- システム・パラメータのアクティブ値 - CMKRNL 特権を持つユーザは、システム管理ユーティリティ (SYSMAN) またはシステム生成ユーティリティ (SYSGEN) を使用して、稼働中 (アクティブ) のシステムのシステム・パラメータを変更できる。ただし、**ダイナミック・システム・パラメータ**として分類されているアクティブ値のみ。
- システム・パラメータの現在値 - SYSPRV 特権を持つユーザは、SYSMAN または SYSGEN を使用して現在のシステムのシステム・パラメータを変更できる。



注意:

システム・パラメータを変更する場合は、1.2 項「パラメータ値の標準的な変更方法」で説明したように、SYSGEN ではなく、できるだけ AUTOGEN または SYSMAN を使用してください。

OPCOM は、現在のシステム・パラメータに対して行われたすべての変更を、SYSGEN メッセージとしてログ・ファイルに記録します。形式は次のとおりです。

```
%%%%%%%%%% %OPCOM <dd-mmm-yyyy hh:mm:ss.cc> %%%%%%%%%%%
Message from user <ユーザ名>
%SYSGEN-I-WRITExxx, <システム・モード> system parameters modified by
process ID
<プロセス id> into file <ファイル指定>
```

例

```
%%%%%%%%%% %OPCOM 3-JUN-2002 08:11:59.55 %%%%%%%%%%%
Message from user D_PLUTO on ANASAT
%SYSGEN-I-WRITECUR, CURRENT system parameters modified by process ID 0000020B
into file SYS$UPDATE:[SYSTEM]UPDATESYS.PAR;2
```

このメッセージは、システム・パラメータの現在値が変更されていることを示しています。



注意:

DCL の SET MESSAGE コマンドを使って、システム・メッセージの形式を変更した場合、そのメッセージはログ・ファイルには示されません。

6.5.2.7 機密保護アラーム・メッセージ

アラーム・メッセージは、選択したイベントが発生したときに機密保護オペレータ・ターミナルに送信されます。機密保護アラーム・メッセージをターミナルで受信できるようにするための方法については、6.6.6 項「ターミナルを使用可能にして、アラーム・メッセージを受信する方法」を参照してください。

例

次の例は、JTQUOTA に変更した後の機密保護アラーム OPCOM メッセージを表しています。

```
%%%%%%%%% OPCOM 6-JAN-2003 10:41:21.10 %%%%%%%%%%
Message from user AUDIT$SERVER on BISCO
Security alarm (SECURITY) and security audit (SECURITY) on BISCO, system id:
20353
Auditable event:          System UAF record modification
Event time:              6-JAN-2003 10:41:20.69
PID:                    00600123
Process name:           SYSTEM
Username:               SYSTEM
Process owner:         [SYSTEM]
Terminal name:         RTA1:
Image name:            BISCO$DUA0:[SYS0.SYSCOMMON.] [SYSEXE] AUTHORIZE.EXE
Object class name:     FILE
Object name:           SYS$SYSTEM:SYSUAF.DAT;4
User record:          NEWPORT
JTQUOTA:              New:          2048
                      Original:    1024
```

6.5.2.8 オペレータ・ログ・ファイルの内容

例 6-2 「オペレータ・ログ・ファイルの例 (SYS\$MANAGER:OPERATOR.LOG)」に、オペレータ・ログ・ファイルに記録される代表的なメッセージの一部を示します。

例 6-2 オペレータ・ログ・ファイルの例 (SYS\$MANAGER:OPERATOR.LOG)

```
%%%%%%%%%% OPCOM, 19-APR-2002 22:26:07.90 %%%%%%%%%%%
Device DMA0: is offline. 1
Mount verification in progress.
%%%%%%%%%% OPCOM, 19-APR-2002 22:26:20.22 %%%%%%%%%%%
Mount verification completed for device DMA0:
%%%%%%%%%% OPCOM, 19-APR-2002 22:33:54.07 %%%%%%%%%%%
Operator '_ZEUS$VT333:' has been disabled, user JONES 2
%%%%%%%%%% OPCOM, 19-APR-2002 22:34:15.47 %%%%%%%%%%%
Operator '_ZEUS$VT333:' has been enabled, user SMITH
%%%%%%%%%% OPCOM, 19-APR-2002 22:34:15.57 %%%%%%%%%%%
operator status for '_ZEUS$VT333:'
PRINTER, TAPES, DISKS, DEVICES
%%%%%%%%%% OPCOM, 19-APR-2002 22:38:53.21 %%%%%%%%%%%
request 1, from user PUBLIC 3
Please mount volume KLATU in device MTA0:
The tape is in cabinet A
%%%%%%%%%% OPCOM, 19-APR-2002 22:39:54.37 %%%%%%%%%%%
request 1 was satisfied.
%%%%%%%%%% OPCOM, 19-APR-2002 22:40:23.54 %%%%%%%%%%%
message from user SYSTEM 4
Volume "KLATU" mounted, on physical device MTA0:
%%%%%%%%%% OPCOM, 19-APR-2002 22:40:38.02 %%%%%%%%%%%
request 2, from user PUBLIC
MOUNT new relative volume 2 () on MTA0:
%%%%%%%%%% OPCOM, 19-APR-2002 22:41:07.54 %%%%%%%%%%%
message from user SYSTEM
Volume "KLATU" dismantled, on physical device MTA0:
15-APR-2002 22:42:14.81, request 2 completed by operator OPA0
%%%%%%%%%% OPCOM, 19-APR-2002 22:46:47.96 %%%%%%%%%%%
request 4, from user PUBLIC
_TTB5:, This is a sample user request with reply expected.
%%%%%%%%%% OPCOM, 19-APR-2002 22:47:38.50 %%%%%%%%%%%
request 4 was canceled
%%%%%%%%%% OPCOM, 19-APR-2002 22:48:21.15 %%%%%%%%%%%
message from user PUBLIC
_TTB5:, This is a sample user request without a reply expected.
%%%%%%%%%% OPCOM, 19-APR-2002 22:49:37.64 %%%%%%%%%%%
Device DMA0: has been write locked.
Mount verification in progress.
%%%%%%%%%% OPCOM, 19-APR-2002 23:33:54.07 %%%%%%%%%%%
message from user NETACP
DECnet shutting down
```

各メッセージの種類は次のとおりです。

- 1 デバイス状態メッセージ
- 2 ターミナルが使用可能または使用不能になったことを示すメッセージ
- 3 ユーザからの要求とオペレータの応答を示すメッセージ
- 4 ボリュームがマウントまたはディスマウントされたことを示すメッセージ

6.5.3 オペレータ・ログ・ファイルの設定

通常、オペレータ・ログ・ファイルはシステム・ディスクの [SYSMGR] というディレクトリに格納されます。しかし、論理名 OPC\$LOGFILE_NAME を定義することによって、ログ・ファイルを別の場所に格納することもできます。

OPERATOR.LOG ファイル (または論理 OPC\$LOGFILE_NAME が指すファイル) のサイズとアクセスは、そのファイルが置かれているディスク・デバイスのサイズとアクセスの制限を受けます。ディスク・デバイスにログ・ファイルを書き込むだけの余裕がなかったり、他の方法での

デバイスへのアクセスが制限されていたりすると、ログ・ファイルから記録が失われる可能性があります。

省略時の設定では、ログ・ファイルは、OpenVMS Cluster 環境のワークステーション (クラスター内の 1 番目のシステムでないワークステーション) を除き、すべてのシステム上に作成されます。OPCOM は、システムにグラフィック・デバイスがあるかテストして、システムがワークステーションかどうかを判断します。具体的には、次のようにテストします。

```
F$DEVICE ("*", "WORKSTATION", "DECW_OUTPUT")
```

論理名 OPC\$LOGFILE_ENABLE に「真」を定義することで、ログ・ファイルが作成されるようになります。

システムをリブートするたびに、OPERATOR.LOG の新しいバージョンが作成されます。各ノードに対して、オペレータ・ログ・ファイルが 1 つ存在します。このファイルは、共用ファイルではありません。

このファイルは ASCII 形式のため、印刷することができます。このファイルを定期的に印刷して参照用に保管してください。6.5.5 項「オペレータ・ログ・ファイルのプリント」に、オペレータ・ログ・ファイルを印刷する方法を説明します。

6.5.3.1 オペレータ・ログ・ファイルの新バージョンの作成

DCL の REPLY/LOG コマンドにより、ファイルの新しいバージョンをいつでも作成することができます。ログ・ファイルとして使用されるのは常に最新のバージョンで、このバージョンは他のユーザがアクセスすることはできません。省略時の設定では、すべてのオペレータ・クラスメッセージがログ・ファイルに記録されます。

次に示すのは、REPLY/LOG コマンドを使用するときのガイドラインです。

- ログ・ファイルに含むオペレータ・クラスを指定するときには、REPLY/LOG/ENABLE=(キーワード) および REPLY/LOG/DISABLE=(キーワード) の 2 つのコマンドが使用できる。
- REPLY/ENABLE コマンドおよび REPLY/DISABLE コマンドで /LOG 修飾子を使用する場合、キーワードで選択したクラスがターミナルで有効あるいは無効になるのではなく、ログ・ファイルへの記録が開始または停止される。

ログ・ファイルがすでにオープンしていると、クラス・リストが保持され、新しく作成されたログ・ファイルで有効になります。ログ・ファイルがオープンしていない場合には、論理名 OPC\$LOGFILE_CLASSES の値が使用されます。この論理名が存在しなければ、新しいログ・ファイルですべてのクラスが有効になります。

OPC\$LOGFILE_CLASSES に不正なクラスが含まれていると、すべてのクラスが有効となり、次のようなメッセージが、システムのスタートアップ時にコンソールに表示され、OPERATOR.LOG ファイルに記録されます。

```
%%%%%%%%%% OPCOM 18-MAY-2002 13:28:33.12 %%%%%%%%%%%  
"BADCLASS" is not a valid class name in OPC$LOGFILE_CLASSES
```

正しいオペレータ・クラスについては、『OpenVMS システム管理者マニュアル (上巻)』を参照してください。

詳細は、『OpenVMS DCL ディクショナリ』の REPLY/LOG, REPLY/ENABLE, REPLY/DISABLE の各コマンドの項目を参照してください。

例

ディスクおよびテープのマウントおよびディスマウント操作を記録するログ・ファイルをオープンします。

```
$ REPLY/LOG/ENABLE= (DISKS, TAPES)
```

6.5.3.2 論理名の指定

コマンド・プロシージャ SYS\$MANAGER:SYLOGICALS.COM に論理名を定義することによって、オペレータ・ログ・ファイルの省略時の状態を指定することができます。次の表に、そのような論理名とその働きをまとめます。SYLOGICALS.COM についての詳細は、『OpenVMS システム管理者マニュアル (上巻)』を参照してください。



重要:

論理名 OPC\$ALLOW_INBOUND および OPC\$ALLOW_OUTBOUND に FALSE を定義すると、大半の OPCOM は、指定された方向に転送されます。返されると想定されている状態メッセージだけでなく、大半の OPCOM メッセージが表示されません。

論理名	働き
OPC\$ALLOW_INBOUND	ノードに戻ってくる OPCOM トラフィックをオフまたはオンする。省略時の設定では、この論理名は、TRUE に定義されている。この論理名を FALSE に定義すると、ノードはクラスタ内の別のノードから大半の OPCOM メッセージを受信しなくなる。
OPC\$ALLOW_OUTBOUND	ノードから出て行く OPCOM トラフィックをオフまたはオンする。省略時の設定では、この論理名は、TRUE に定義されている。この論理名を FALSE に定義すると、ノードはクラスタ内の別のノードへ大半の OPCOM メッセージを送信しなくなる。
OPC\$LOGFILE_ENABLE	オペレータ・ログ・ファイルを開くかどうかを指定する。この論理名を TRUE に定義するとオペレータ・ログ・ファイルを開き、FALSE に定義すると開かない。省略時の設定では、OpenVMS Cluster 環境上のワークステーションを除くすべてのシステム上でログ・ファイルが開く。
OPC\$LOGFILE_CLASSES	ログ・ファイルに記録するイベントのオペレータ・クラスを指定する。省略時の設定では、すべてのクラスのイベントを記録するものとしてログ・ファイルを開く。論理名は、適用するクラスの検索リスト、コマンドで区切ったリスト、あるいはその両方の組み合わせで指定できる。OPC\$LOGFILE_ENABLE を定義しない場合でも OPC\$LOGFILE_CLASSES を定義できる。その場合、指定したクラスは、開くすべてのログ・ファイルに使用される。しかし、各ログ・ファイルを開くかどうかは省略時の設定が適用される。有効なオペレータ・クラスについての説明は、『OpenVMS システム管理者マニュアル (上巻)』を参照。
OPC\$LOGFILE_NAME	ログ・ファイルの名前を指定する。この論理名の定義を省略すると、ログ・ファイルの名前は SYS\$MANAGER:OPERATOR.LOG となる。システム・ディスク以外のディスクを指定する場合は、コマンド・プロシージャ SYLOGICALS.COM にそのディスクをマウントするコマンドを加える必要がある。
OPC\$OPAO_ENABLE	クラスタ内のワークステーション用のシンボル値を上書きする。この論理名を TRUE と定義すると、OPAO デバイスを BROADCAST (NOBROADCAST の省略時の設定を上書き) に設定する。クラスタ内のワークステーションではないシステムの場合、この論理名を FALSE と定義すると、OPAO デバイスが NOBROADCAST に設定される。



注意:

OPCOM の初期のスタートアップ以外でも使用される論理名は OPC\$LOGFILE_NAME だけです。他の OPCOM 論理名は無視されます。たとえば、論理名 OPC\$LOGFILE_ENABLE が FALSE に定義されていても、REPLY/LOG コマンドを実行すれば新しいオペレータ・ログ・ファイルがオープンします。OPCOM のスタートアップ後にその状態とクラスを再設定するためには、REPLY/ENABLE コマンドまたは REPLY/DISABLE コマンドを使用します。

6.5.4 オペレータ・ログ・ファイルの管理

オペレータ・ログ・ファイルを定期的に管理するための計画をたててください。まず、毎日新しいログ・ファイルを起動し、前日に使用していたファイル(2番目に新しいバージョン)をリネームするという方法があります(次の項の例を参照)。あるいは、古いファイルを削除することもできます。ただし、ログ・ファイルを削除する場合には、必ずそのバックアップをとるようにしてください。詳細は『OpenVMS システム管理者マニュアル(上巻)』を参照してください。

OPCOM を誤って削除してしまった場合は、次の手順に従って手動で起動します。

1. SYSTEM アカウントにログインして、この操作に必要な特権を入手する。
2. 次のコマンドを入力して、スタートアップ・コマンド・プロシージャ (STARTUP.COM) を実行し、コマンド・パラメータとして OPCOM を指定する。

```
$ @SYS$SYSTEM:STARTUP OPCOM
```

6.5.5 オペレータ・ログ・ファイルのプリント

次に、オペレータ・ログ・ファイルの最新のバージョンをプリントする手順を示します。この作業を行うためには、OPER 特権が必要です。

1. ターミナルをオペレータ・ターミナルとして宣言 (使用可能に) する。

```
$ REPLY/ENABLE
```

2. 現在のログ・ファイルをクローズし、新しいファイルをオープンする。

```
$ REPLY/LOG
```

3. 省略時の値を SYS\$MANAGER に設定し、次のコマンドによってファイルのすべてのバージョンを表示する。

```
$ SET DEFAULT SYS$MANAGER  
$ DIRECTORY OPERATOR.LOG
```

4. 2番目に新しいバージョンを OPERATOR.OLD にリネームする。

```
$ RENAME OPERATOR.LOG;-1 OPERATOR.OLD
```

バージョン番号 -1 は、このファイルの 2 番目に新しいバージョンを表す。なお、最も大きなバージョン番号は、現在使用中のオペレータ・ログ・ファイルである。

5. オペレータ・ログ・ファイルをプリントする。

```
$ PRINT OPERATOR.OLD
```

例

```
$ REPLY/ENABLE 1  
$ REPLY/LOG 2
```

```

%%%%%%%%%%%% OPCOM, 19-APR-2002 12:28:20.11 %%%%%%%%%%%%%
Logfile was closed by operator _MARS$VTA2: 3
Logfile was HOMER::SYS$MANAGER:[SYSMGT] OPERATOR.LOG;27
%%%%%%%%%%%% OPCOM, 19-APR-2002 12:29:24.52 %%%%%%%%%%%%%
Logfile has been initialized by operator _MARS$VTA2:
Logfile is HOMER::SYS$MANAGER:[SYSMGT] OPERATOR.LOG;28

```

```

$ SET DEFAULT SYS$MANAGER 4
$ DIRECTORY OPERATOR.LOG 5

```

Directory SYS\$MANAGER:[SYSMGT]

```

OPERATOR.LOG;28 OPERATOR.LOG;27
Total of 2 files.
$ RENAME OPERATOR.LOG;-1 OPERATOR.OLD 6
$ PRINT OPERATOR.OLD 7

```

番号が付いた各行の意味は次のとおりです。

- 1 REPLY/ENABLE コマンドにより、ターミナルをオペレータ・ターミナルとして宣言する。
- 2 REPLY/LOG コマンドにより、現在のログ・ファイルをクローズし、新しいファイルをオープンする。
- 3 新しいログ・ファイルがオープンされたことを示す OPCOM からのメッセージ。
- 4 SET DEFAULT コマンドにより、オペレータの省略時のディスクとしてシステム・ディスクを使用するよう設定する。
- 5 DIRECTORY コマンドにより、システム・ディスク上の [SYSMGR] ディレクトリ内のファイルを表示する。
- 6 RENAME コマンドにより、オペレータ・ログ・ファイルの 2 番目に新しいバージョンを OPERATOR.OLD にリネームする
- 7 PRINT コマンドにより、古いオペレータ・ログ・ファイル OPERATOR.OLD をプリントする。

6.6 機密保護監査機構の使用法

この節では、機密保護監査機構の働き、機密保護監査機構の起動、および機密保護監査ログ・ファイルを新しく作成する方法を説明します。機密保護監査ログ・ファイルについての詳細は、『OpenVMS システム・セキュリティ・ガイド』を参照してください。

6.6.1 機密保護監査機構について

機密保護監査機構は、機密保護関係のイベントがシステム上で発生したときに、それを記録する機能です。機密保護関係のイベントは、**イベント・クラス**と呼ばれるカテゴリに分類されません。

省略時の設定では、システムを 表 6-7 「省略時のイベント・クラス」に示すイベント用にインストールまたはアップグレードしたときに、機密保護監査機構が使用できるようになります。

表 6-7 省略時のイベント・クラス

クラス	説明
ACL	機密保護監査機構 ACE を持つ全オブジェクトへのアクセス。
AUDIT	SET AUDIT コマンドの全用途。このカテゴリは使用禁止にできない。
AUTHORIZATION	登録データベースに加えたすべての変更。 <ul style="list-style-type: none"> • システム利用者登録ファイル (SYSUAF) • ネットワーク代理登録ファイル (NETPROXY および NET\$PROXY) • ライト・データベース (RIGHTSLIST)

表 6-7 省略時のイベント・クラス (続き)

クラス	説明
BREAKIN	すべてのブレイクインの試み。バッチ、独立、ダイアルアップ、ローカル、ネットワーク、遠隔。
LOGFAILURE	すべてのログイン障害。バッチ、ダイアルアップ、ローカル、遠隔、ネットワーク、サブプロセス、独立。

使用しているサイトにおける機密保護の必要条件が、その他の監査にも合う場合は、6.6.4 項「その他のクラスに対して機密保護監査機構を使用する方法」で説明するように、DCL の SET AUDIT コマンドを使用して、別のイベント・クラスを使用可能にすることができます。

6.6.1.1 機密保護監査ログ・ファイル

監査サーバ・プロセスは、システム起動時に作成され、機密保護監査ログ・ファイル SYS\$MANAGER:SECURITY.AUDIT\$JOURNAL 中の特定のイベントを記録します (記録されるイベントについては、表 6-7 「省略時のイベント・クラス」を参照)。

定期的にファイルを検討するときの手順によって、機密保護監査ログ・ファイルの有用性は変わってきます。サイトの監査検討方針の一部として、たとえば次のような手順が考えられます。

1. 毎朝、機密保護管理ログ・ファイルの新しいバージョンを作成する。
2. 前日のログ・ファイルを見て、問題があると思われるシステム・アクティビティがないか、検討する。システムについて監査する機密保護イベントの数によっては、監査ログ・ファイルに書き込まれる監査レコードすべてを検討することは現実的ではない場合がある。そのようなときには、特定のレコード・セット (たとえば、Authorization レコードや Breakin レコード、あるいは通常の勤務時間外に作成された全イベント) をログ・ファイルから選択したい場合もある。
3. 検討中に、問題があると思われる機密保護イベントが見つかった場合は、機密保護監査ログ・ファイルをより詳細に調べる (『Security Guide』を参照)。

6.6.1.2 混合バージョン・クラスタの監査ログ・ファイル

以前のバージョンのシステムで実行される監査分析ユーティリティ (ANALYZE/AUDIT) は、最新バージョンの監査ログ・ファイルを処理できません。最新バージョンを処理するには、ANALYZE/AUDIT の最新バージョンを使用する必要があります。混合バージョンのクラスタでは、別々の監査ログ・ファイルを保守することをお勧めします。

監査ログ・ファイルの出力先を変更するには、以前のバージョンを実行するノードと最新バージョンを実行するノードの両方で、次のコマンドを発行します。

AUDIT/JOURNAL/DESTINATION=ファイル指定

ここで指定したファイル指定は、監査サーバ・データベース・ファイルに格納されます。省略時の設定では、このファイルは SYS\$COMMON:[SYSMGR] に格納され、それぞれ SECURITY_AUDIT.AUDIT\$JOURNAL と SECURITY.AUDIT\$JOURNAL と呼ばれます。

オペレーティング・システムは、ワークステーションと制限された管理リソースを持つユーザが、監査ログ・ファイルを別のノードに複製することを許可します。2 次ログ、つまり機密保護アーカイブ・ファイルは、ファイルを解析できる遠隔ノード上のセキュリティ・アドミニストレータが使用できます。

クラスタ内の各ノードは、各自のアーカイブ・ファイルを持っていない限りなりません。アーカイブ・ファイルは、クラスタ内の複数のノードでは共用できません。

詳細は『OpenVMS システム・セキュリティ・ガイド』を参照してください。

6.6.2 機密保護監査情報の表示

現在サイトが監査しているイベント・クラスを調べるには、DCLのSHOW AUDIT コマンドを入力します。

表示される機密保護情報の例を次に示します。

```
$ SHOW AUDIT
```

```
System security alarms currently enabled for:
```

```
ACL
Breakin:          dialup,local,remote,network,detached
Privilege use:
  SECURITY
Privilege failure:
  SECURITY
```

```
System security audits currently enabled for:
```

```
ACL
Authorization
Breakin:          dialup,local,remote,network,detached
Login:            dialup,local,remote,network,detached
Logfailure:       batch,dialup,local,remote,network,subprocess,detached
Logout:           dialup,local,remote,network,detached
Privilege use:
  SECURITY
```

```
Privilege failure:
```

ACNT	ALLSPOOL	ALTPRI	AUDIT	BUGCHK	BYPASS	CMEXEC	CMKRNL
DETACH	DIAGNOSE	EXQUOTA	GROUP	GRPNAM	GRPPRV	LOG_IO	MOUNT
NETMBX	OPER	PFNMAP	PHY_IO	PRMCEB	PRMGBL	PRMMBX	PSWAPM
READALL	SECURITY	SETPRV	SHARE	SHMEM	SYSGBL	SYSLCK	SYSNAM
SYSPRV	TMPMBX	VOLPRO	WORLD				

```
DEVICE access:
```

```
Failure:          read,write,physical,logical,control
```

```
FILE access:
```

```
Failure:          read,write,execute,delete,control
```

```
VOLUME access:
```

```
Failure:          read,write,create,delete,control
```

6.6.3 監査の開始を遅らせる方法

通常は、SYSTARTUP_VMS.COM が実行される直前に VMS\$LPBEGIN の監査が開始されますが、論理名 SYS\$AUDIT_SERVER_INHIBIT を定義し直せば、この動作を変更することができます。

オペレーション・システムが機密保護イベント・メッセージを送り始めるタイミングを変更するには、次の行を SYS\$STARTUP:SYLOGICALS.COM コマンド・プロシージャに追加します。

```
$ DEFINE/SYSTEM/EXECUTIVE SYS$AUDIT_SERVER_INHIBIT YES
```

これで、システム・スタートアップの別のフェーズ(おそらく、SYSTARTUP_VMS.COM の終わり)で監査を開始することができます。これを行うには、コマンド・ファイルを編集して、次の行を追加します。

```
$ SET AUDIT/SERVER=INITIATE
```

SYSTARTUP_VMS.COM の編集に関しては、『OpenVMS システム管理者マニュアル (上巻)』を参照してください。

6.6.4 その他のクラスに対して機密保護監査機構を使用する方法

表 6-7 「省略時のイベント・クラス」に示したクラス以外のクラスに対して機密保護監査を行うには、次の形式を使用します。

```
SET AUDIT/ENABLE= キーワード [...] {/ALARM | /AUDIT}
```

使用可能にできるイベント・クラスの説明については、『OpenVMS システム・セキュリティ・ガイド』を参照してください。

その他のイベント・クラスを監査できるようにするためには、次の2つの修飾子を指定しなければなりません。

1. /ENABLE
2. /ALARM または /AUDIT のいずれか (必ず1つは指定すること。また、両方とも指定してもかまわない。)

/ENABLE, /ALARM, /AUDIT の各修飾子について、次に説明します。

修飾子	説明
/ENABLE	監査するイベント・クラスを定義する。詳細は 第7章 「資源使用状況の調査」を参照。
/ALARM, /AUDIT	イベント・メッセージのデスティネーションを定義する。 <ul style="list-style-type: none">• /ALARM は、メッセージのデスティネーションを使用可能になっている全セキュリティ・オペレータ・ターミナルに指定する。• /AUDIT は、メッセージのデスティネーションを機密保護ログ・ファイルに指定する。 重要なイベントを報告するには、/ALARM 修飾子と /AUDIT 修飾子を使用する。比較的重要なでないイベントは、後で調べることができるように機密保護監査ログ・ファイルだけに書き込んでおくことができる。 表 6-7 「省略時のイベント・クラス」に示す省略時のイベント・クラスは、ALARM および AUDIT として送られる。

新しいイベントの監査は、全ノードでそれが使用できるようにすると、すぐに開始されます。

例

1. 次のコマンドを実行すると、ボリュームのマウントとディスマウントに対する監査が使用可能になります。さらに、メッセージを機密保護監査ログ・ファイルに送信します。

```
§ SET AUDIT/ENABLE=MOUNT/AUDIT
```

2. 次のコマンドを実行すると、ファイルへのアクセス失敗に対する監査が可能になります。さらに、メッセージを機密保護監査ログ・ファイルおよびすべての使用可能なセキュリティ・オペレータ・ターミナルに送信します。

```
§ SET AUDIT/ALARM/AUDIT/ENABLE=ACCESS=FAILURE/CLASS=FILE
```

6.6.5 機密保護監査機構の使用禁止

ユーザが次の構文を使って明示的に /DISABLE 修飾子が指定されたクラスを使用禁止にするまで、監査は続けられます。

```
SET AUDIT/DISABLE= キーワード [...] {/ALARM | /AUDIT}
```

6.6.6 ターミナルを使用可能にして、アラーム・メッセージを受信する方法

アラーム・メッセージは、セキュリティ・クラス・メッセージ用に使用可能にされているターミナルに送信されます。機密保護アラーム・メッセージはオペレータ・ログ・ファイルに書き込まれず、セキュリティ・クラス・メッセージ用に使用可能にされているターミナルだけに表示されます。

多くの場合、機密保護アラーム・メッセージは省略時の設定としてシステム・コンソールに表示されますが、メッセージは画面上を高速でスクロールするので、セキュリティ・クラス・メッセージ用に別個のターミナルを使用できるようにしておき、システム・コンソールにはメッセージが表示されないようにしておくといよいでしょう。

安全な場所にあるターミナルをハードコピーの出力用として指定しておくか、あるいはセキュリティ・オペレータ・ターミナルを監視する専門の担当者を決めておくようにしてください。セキュリティ・オペレータとして使用可能にするターミナルの数に制限はありません。

セキュリティ・クラス・アラーム・メッセージを受信するようにターミナルを設定するには、指定したターミナルで次の DCL コマンドを入力します。

```
$ REPLY/ENABLE=SECURITY
```

例

次に、機密保護アラーム・メッセージの例を示します。

```
%%%%%%%%% OPCOM 25-MAY-2002 16:07:09.20 %%%%%%%%%%
Message from user AUDIT$SERVER on GILMORE
Security alarm (SECURITY) on GILMORE, system id: 20300
Auditable event:          Process suspended ($SUSPND)
Event time:              25-MAY-2002 16:07:08.77
PID:                    30C00119
Process name:           Hobbit
Username:                HUBERT
Process owner:          [LEGAL,HUBERT]
Terminal name:          RTA1:
Image name:              $99$DUA0: [SYS0.SYSCOMMON.] [SYSEXE] SET.EXE
Status:                  %SYSTEM-S-NORMAL, normal successful completion
Target PID:              30C00126
Target process name:     SMISERVER
Target username:         SYSTEM
Target process owner:    [SYSTEM]
```

6.6.7 機密保護レポートの作成

作成するレポートの最も一般的なタイプは、簡略レポートで、これはイベントの日誌リストです。その日の機密保護イベント・メッセージのレポートを作成し、そのレポートを MAIL を使ってシステム管理者に送信するためのコマンド・プロシージャを作成して、毎夜バッチ・ジョブ形式で実行することができます。



注意:

MOUNT コマンドは、監査レコードの /NOLABEL を /FOREIGN に変換するので、ANALYZE/AUDIT/SELECT=MOUNT_FLAGS=NOLABEL の代わりに ANALYZE/AUDIT/SELECT=MOUNT_FLAGS=FOREIGN を使用してください。

次に、ANALYZE/AUDIT コマンド行を使用して、このようなレポートを作成する例を示します。

```
$ ANALYZE/AUDIT/SINCE=TODAY/OUTPUT=31JAN2002.AUDIT -
_$ SYS$MANAGER:SECURITY.AUDIT$JOURNAL
$ MAIL/SUBJECT="Security Events" 31JAN2002.AUDIT SYSTEM
```

6.6.8 機密保護監査ログ・ファイルの新しいバージョンの作成

ユーザ側で何らかの処置を行わない限り、機密保護監査ログ・ファイルは大きくなり続けるので、その保守には工夫が必要です。

クラスタ用機密保護監査ログ・ファイルを新しく作成するためには、SET AUDIT コマンドを入力します。それまでに記録された監査メッセージが失われないように、メモリ内に記憶されたすべての監査メッセージがファイルに書き込まれるまで、監査ログ・ファイルの古いバージョンはクローズされません。

6.6.8.1 ログ・ファイルの新しいクラスタ全体としてのバージョンの作成

機密保護監査ログ・ファイルの、新しい、クラスタ全体としてのバージョンを作成するには、次のコマンドを使用します。

```
$ SET AUDIT/SERVER=NEW_LOG
```

監査サーバ・プロセスにより、クラスタ・ノードごとに監査ログ・ファイルの新しいバージョンがオープンされます。

新しいログをオープンしたら、古いバージョンをリネームします。これには、データの開始または終了日付をファイル名に組み込む、ファイルの命名規則を使用します。次に古いログを別のディスクにコピーし、ディスク空間を節約するためシステム・ディスクからこのログを削除します。そして、古いログに対して監査分析ユーティリティを実行します。

このファイルを保管しておくことにより、クラスタ全体としての監査メッセージの履歴を管理します。システム上に機密保護の侵害の恐れがあった場合、指定した時間帯に保管したログ・ファイルを分析して、疑わしいユーザ・アクティビティを追跡することができます。

6.6.8.2 ログ・ファイルの新しいノード固有のバージョンの作成

場合によっては、OpenVMS Cluster ノードが同じシステム・セキュリティ監査ログ・ファイルを共用していないことがあります。機密保護監査ログ・ファイルの新しい、ノード固有のバージョンを作成するには、次のコマンドを使用します。

```
$ SET AUDIT/DESTINATION= ファイル指定
```

```
$ SET AUDIT/SERVER=NEW_LOG
```

ファイル指定には、ノード固有のファイルを指す論理名 (SYS\$SPECIFIC:[SYSMGR]SECURITY など) を指定します。別のノード上のシステム機密保護監査ログ・ファイルには影響しません。

6.7 オペレーティング・システムの性能の監視

Monitor ユーティリティ (MONITOR) は、オペレーティング・システムの性能に関する情報を入手するためのシステム管理ツールです。さまざまな MONITOR 修飾子を指定することにより、稼働中のシステムからシステム性能データを収集したり、以前にレコード・ファイルに記録されたデータをプレイバックしたりすることができます。プレイバックしたデータは、表示、要約したり、記録し直してレコード・ファイル内のデータ量を減らしたりすることもできます。

この節では、MONITOR ユーティリティの働きと、MONITOR ユーティリティによって情報を表示したり記録したりするいくつかの異なる方法を説明します。具体的には、次のトピックを取り上げます。

作業	参照箇所
Monitor ユーティリティの起動	6.7.2 項 「MONITOR の起動」
システムの動作の表示	6.7.3 項 「システムの動作の表示」
システムの動作の記録	6.7.4 項 「システムの動作の記録」
システムの動作の表示と記録	6.7.5 項 「システムの動作の表示と記録」
記録した動作のプレイバック	6.7.6 項 「記録した動作のプレイバック」
記録した動作の遠隔プレイバック	6.7.7 項 「記録した動作の遠隔プレイバック」
記録ファイルの更新	6.7.8 項 「記録ファイルの更新」
MONITOR の継続実行	6.7.9 項 「MONITOR の継続実行」
遠隔監視	6.7.10 項 「遠隔監視」

Monitor ユーティリティにより得られる情報の解釈については、『Guide to OpenVMS Performance Management』を参照してください。また、Monitor ユーティリティの使用方法については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

6.7.1 MONITOR について

MONITOR を使用して、システム全体の性能データ (システム入出力統計、ページ管理統計、各プロセッサ・モードの動作時間など) を特定の間隔で監視し、いろいろな形式で出力することができます。また、MONITOR をバックグラウンド・プロセスとして継続的に実行することにより、システムの性能情報のデータベースを開発することもできます。これについては、6.7.9 項「MONITOR の継続実行」を参照してください。

6.7.1.1 MONITOR クラス

各 MONITOR クラスはいくつかのデータ項目から構成されます。個々のクラスに対して定義されているデータ項目の一覧については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の MONITOR コマンドの説明を参照してください。

特別な情報のクラスを監視するには、MONITOR コマンド行にクラス名を指定します。MONITOR が表示する情報は、選択したクラスのタイプによって異なります。表 6-8 「MONITOR クラスの種類」では、2 つの MONITOR クラス・タイプの比較を示します。

表 6-8 MONITOR クラスの種類

クラスの種類	説明
システム・クラス	システム全体の資源の使用状況に関する統計値。
コンポーネント・クラス	個々のコンポーネントがシステムあるいはクラスタ全体に対してどれだけ「貢献」しているかを示す統計値。

MONITOR クラスの 2 つの種類の違いとして、IO クラスと DISK クラスを例にとりて考えることができます。IO クラスは、システム全体のすべての直接入出力操作を計測するデータ項目が含まれるため、システム・クラスに属します。一方、DISK クラスは、個々のディスクの直接入出力操作を計測するため、コンポーネント・クラスに属します。

表 6-9 「MONITOR クラス」に、各 MONITOR クラスとクラスの種類 (システムかコンポーネントか) を示します。

表 6-9 MONITOR クラス

クラス	種類	説明
ALL_CLASSES	システムまたはコンポーネント	すべてのクラスの統計値
CLUSTER	システム	クラスタ全体の性能に関する統計値
DECNET	システム	DECnet for OpenVMS に関する統計値
DISK	コンポーネント	ディスク入出力に関する統計値
DLOCK	システム	分散型ロック管理情報の統計値
FCP	システム	ファイル制御プリミティブに関する統計値
FILE_SYSTEM_CACHE	システム	ファイル・システム・キャッシュに関する統計値
IO	システム	システム入出力に関する統計値
LOCK	システム	ロック管理情報の統計値
MODES	コンポーネント	各プロセッサ・モードでの動作時間
MSCP_SERVER	システム	MSCP サーバに関する統計値

表 6-9 MONITOR クラス (続き)

クラス	種類	説明
PAGE	システム	ページ管理情報の統計値
PROCESSES	コンポーネント	すべてのプロセスに関する統計値
RMS	コンポーネント	レコード管理サービス (RMS) に関する統計値
SCS	コンポーネント	システム通信サービスに関する統計値
STATES	システム	スケジューラ状態ごとのプロセス数
SYSTEM	システム	他のクラスに関する統計値の要約
TRANSACTION	システム	DECdtm サービスに関する統計値
VBS ¹	システム	仮想バランス・スロットに関する統計値
VECTOR	システム	スケジューリングされたベクタ・プロセッサの使用

1 VAX のみ

6.7.1.2 データの表示

PROCESSES クラスのものを除き、表示可能なデータ項目はすべてレートとレベルで表されません。

- レートは、秒あたりの発生数
- レベルは、監視されたデータ項目のサイズを示す値

データ項目ごとに、次の 4 種類の統計値をどれでも、また何種類でも要求することができます。

統計値	説明
レートまたはレベルの現在値	最も新しく収集された、レートまたはレベルの値
レートまたはレベルの平均値	MONITOR 要求の最初から測定される
レートまたはレベルの最小値	MONITOR 要求の最初から測定される
レートまたはレベルの最大値	MONITOR 要求の最初から測定される

DISK, MODES, SCS, STATES の各クラスの場合は、オプションとしてすべての統計値をパーセンテージで表すことができます。

PROCESSES クラスでは、MONITOR は、説明情報、レベル情報、および時間の経過で増加するカウンタが表示されます。

6.7.1.3 出力のタイプ

MONITOR は、システム性能データをクラスごとに収集し、指定した修飾子によって、次のように 3 種類のオプションの形式で出力します。

修飾子	説明
/DISPLAY	ASCII 画面イメージ形式の出力を生成する。これは /VIEWING_TIME 修飾子により指定される頻度で作成される。
/RECORD	要求されたクラスのために収集したデータを含むバイナリ・レコード・ファイルを生成する。インターバルごとに、各クラスに 1 つのレコードが作成される。
/SUMMARY	MONITOR 要求の間に要求された全クラスの要約統計値を含む ASCII ファイルを生成する。

上記の修飾子のいずれかとともに /INPUT を指定すると、MONITOR は、以前に作成したレコード・ファイルから 1 つまたは複数の性能データを収集します。そうでない場合は、データはカウンタと稼働システム上のデータ構造から収集されます。

MONITOR 要求を開始したい場合には /BEGINNING 修飾子を、終了したい場合には /ENDING 修飾子をそれぞれ使用します。

/DISPLAY 修飾子の使用方法

MONITOR により収集された情報は、通常は ASCII 画面イメージとして表示されます。/DISPLAY 修飾子を使用すると、ディスク・ファイルにこの情報を含めるようにオプション指定することができます。ファイル指定を省略すると、出力先は SYS\$OUTPUT になります。



注意:

/DISPLAY 修飾子を使用する場合は注意が必要です。MONITOR は継続的に表示情報をファイルに入力するため、短時間にそのサイズが大きくなっていくためです。

/DISPLAY 修飾子については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

/RECORD 修飾子の使用方法

/RECORD 修飾子を使用すると、クラスに関する全データが記録されます。これは、単一の統計値や単一のコンポーネント統計値クラスの項目だけを同時に表示している場合でも同じです。このファイルは MONITOR 要求が開始されたときに作成され、要求が終了するとクローズします。結果として得られたファイルを今後の要求のソース・ファイルとして使用して、ターミナル上でデータを形式化して表示したり、要約ファイルを作成したり、別の特性を持つ新しいレコード・ファイルを作成したりすることができます。

6.7.2 MONITOR の起動

Monitor ユーティリティを起動するためには、次の DCL コマンドを入力します。

```
$ MONITOR
```

次のプロンプトが表示されます。

```
MONITOR>
```

このプロンプトに対して、任意の MONITOR コマンドを入力することができます。詳細は『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。ただし、最もよく使用する MONITOR コマンドは、クラス名を指定します。

例

```
MONITOR> MONITOR PAGE
```

この例は、ページ管理情報の統計値を監視するために MONITOR コマンドで PAGE クラスを指定しています。

DCL コマンド・レベルからも MONITOR のコマンドを使用することができます。

MONITOR 要求の変更または終了

MONITOR コマンドによる要求の動作は、/ENDING 修飾子を指定するまで続けられます。ただし、MONITOR 要求を変更したり終了したりする場合は、次のいずれかを押します。

キー	説明
Ctrl/W	/VIEWING_TIME 値を一時的に変更し、現在の画面の直後に新しい画面を生成する。この機能は、ブロードキャスト・メッセージが MONITOR 表示領域を上書きしてしまった場合に便利である。 また、/VIEWING_TIME の値が大きいときに Ctrl/W を使うと、要求があり次第表示イベントを生成することができる。
Ctrl/C	現在の要求を終了するが、ユーティリティは終了しない。したがって、引き続き MONITOR> プロンプトから新しい要求を開始したり、任意の MONITOR コマンドを入力することができる。
Ctrl/Z	現在の要求を終了して、かつ MONITOR も終了する。

6.7.3 システムの動作の表示

システムの動作の表示モードは、定期的に、あるいはインストールのチェック、チューニング、トラブルシューティングで、稼働中のシステムの動作をリアルタイムに調べたい場合に使用します。出力の履歴情報は記録されません。次の例は、システムの動作の表示モードの使用方法を示しています。

例

1.

```
$ MONITOR PROCESSES/TOPCPU
```

前回の表示からこのコマンドを実行するまでに CPU を最も使用した 8 つのプロセスを示す棒グラフが表示される。また、各プロセスが使用した CPU 時間も表示される。

このコマンドにより次のような出力が生成される。

```
OpenVMS Monitor Utility
TOP CPU TIME PROCESSES
on node BOMBAY
20-JAN-2002 10:06:49

0          25          50          75          100
+ - - - - + - - - - + - - - - + - - - - +
07E00181  CAFARET      100 *****
|          |          |          |          |
|          |          |          |          |
|          |          |          |          |
|          |          |          |          |
|          |          |          |          |
|          |          |          |          |
|          |          |          |          |
+ - - - - + - - - - + - - - - + - - - - +
```

この例では、ユーザ CAFARET が使用できる CPU 時間を 100 パーセント使用している。ユーザが使用しているコンピュータの資源についてより詳細な情報を表示するには、次のようなコマンドを使用する。

```
$ SHOW PROCESS/CONTINUOUS/ID=07E00181
```

この例では、結果として得られる表示の中で最も役に立つ情報は、イメージの名前であり、たとえば次のように、最後の部分に表示される。

```
.  
. .  
$1$DUA1: [SYS1D.SYSCOMMON.] [SYSEXE] RODAN.EXE
```

この例は CAFARET が RODAN.EXE を実行していることを示している。これは新しいソフトウェアであり、その動作がループしている可能性がある。このような状況は、CAFARET が特権ユーザで、別のユーザよりも高い優先順位でプロセスを実行した場合に発生する。

2.

```
$ MONITOR/DISPLAY=PROCESSES.LOG PROCESSES
```

MONITOR からの情報は、サポートされている任意のターミナルまたはディスク・ファイルに出力することができる。ここでは、MONITOR のプロセス統計を PROCESSES.LOG ファイルに書き込んでいる。この後、このファイルをハードコピー・デバイスに出力してプリントすることができる。



重要:

データは継続的に表示ファイルに追加されるため、ファイルが大きくなりすぎないように注意してください。

3.

```
$ MY_CLASSES ::= -  
_ $ "DECNET+FCP+IO+LOCK+MODES+PAGE+PROCESSES+STATES"  
$ MONITOR/NODE=(CURLEY,LARRY)/INTERVAL=20/VIEWING_TIME=8 'MY_CLASSES'
```

頻繁に使用するクラス名の組み合わせは、DCL シンボルに定義しておくとな便利なことがある。ここでは、CURLEY および LARRY という OpenVMS Cluster ノード に関して選択されたクラスのデータが 20 秒ごとに収集される。また、クラスのうちの 1 つに関して収集されたデータのうちの最新の項目が 8 秒ごとに表示される。MONITOR では、クラスの表示順序があらかじめ決められている。

6.7.4 システムの動作の記録

システムの動作の記録は、将来のために MONITOR データをとっておく必要がある場合に使用します。次のような用途が考えられます。

- インストールのチェック、チューニング、トラブルシューティング。すなわち、システムの動作の表示で示したすべての用途。
ターミナルに物理的に表示できる量より多くの情報を入手したい場合、ターミナルが利用できない場合、またはシステムのデータを入手する必要があるがしばらくはデータ収集のための時間がとれないという場合、記録モードを利用できる。
- 長期間に渡って定期的に性能のデータを収集する。
MONITOR データを定期的に記録してまとめることで、システム資源の使用量に関する長期間のデータを収集することができる。



重要:

データは継続的に表示ファイルに追加されるため、ファイルが大きくなりすぎないように注意してください。

次の例は、システムの動作の記録モードの使用方を示しています。

例

```
$ MONITOR/NODE=(LARRY,MOE)/NODISPLAY/RECORD MODES+STATES
```

各プロセッサ・モードでの動作時間、および LARRY および MOE というノードの各スケジュール状態におけるプロセス数のデータが記録されます。ただし、この情報は出力されません。

6.7.5 システムの動作の表示と記録

システムの動作の表示と記録モードは、性能データを保持し、収集されるときにその性能データを表示させる場合に使用します。MONITOR では、記録ファイルに共用読み込みアクセス権が設定されるので、他の表示プロセスが記録ファイルを書き込んでいる間に、それを別の表示プロセスがプレイバックすることができます。

次の例は、性能の情報を記録しながら、同時に収集中のデータを表示する方法を示しています。最初の例では、データの収集と記録の両方を同じコマンドで行います。2 番目および 3 番目の例は、2 つの別々のプロセスを使って記録と表示を同時に行う方法を示しています。2 番目の例のプロセスは記録を行い、3 番目の例のプロセスがファイルをプレイバックして要約します。

例

1.

```
$ MONITOR/RECORD FCP/AVERAGE,FILE_SYSTEM_CACHE/MINIMUM
```

ファイル・システムとファイル・システム・キャッシュのデータを 3 秒ごとに収集して記録する。さらに、棒グラフによって、FCP の平均値と FILE_SYSTEM_CACHE の最小値が表示される。2 つのグラフが 3 秒ごとに交互に表示される。現在の統計値は次のプレイバック要求で得られる。

2.

```
$ MONITOR/RECORD=SYS$MANAGER:ARCHIVE.DAT -  
_$_ /INTERVAL=300/NODISPLAY ALL_CLASSES
```

すべてのクラスのデータを同時に 5 秒間隔で保存する。同様のコマンドをバッチ・ジョブで実行し、ディスクの使用量を注意深く監視すると便利である。

3.

```
$ MONITOR/INPUT=SYS$MANAGER:ARCHIVE.DAT: -  
_$_ /NODISPLAY/SUMMARY/BEGINNING="-1" PAGE,IO
```

報告された性能の問題の調査の一部として、過去 1 時間に発生したページ動作および入出力動作をまとめる。記録を行うプロセスは 5 分ごとに OpenVMS RMS フラッシュ動作を行うため、過去 5 分以内の収集データは表示を行うプロセスからは利用できない点に注意。

/FLUSH_INTERVAL 修飾子により、フラッシュ動作を行う間隔を明示的に指定できる。表示を行うプロセスには、記録ファイルに対する読み込みアクセス権が必要。

6.7.6 記録した動作のプレイバック

プレイバックとは、記録ファイルに収集されたデータの全部または一部をターミナルに表示したり、要約レポートとしてまとめたりすることをいいます。データは、クラス、ノード、あるいは時間帯に基づいてまとめることができます。たとえば、24 時間に渡っていくつかのクラスのデータを収集した場合、その間の任意の時間帯の 1 つ以上のクラスのデータを調べたり、まとめたりできます。

記録した時間帯と別の時間帯のデータを表示したりまとめたりすることも可能です。スクリーンに表示を行う実際の間隔は、/VIEWING_TIME 修飾子で制御します。次の例は、記録した動作のプレイバックを行う方法を示しています。

例

1.

```
$ MONITOR/RECORD/INTERVAL=5 IO
```

\$ MONITOR/INPUT IO

システム入出力の統計をとる。最初のコマンドは、データの収集と表示を 5 秒間隔で行う。この作業は、このコマンドを入力した時点から Ctrl/Z を押すまで続けられる。さらにこのコマンドは、省略時の出力ファイル MONITOR.DAT にバイナリ・データを記録する。

2 番目のコマンドは、MONITOR.DAT のデータを入力データとして、入出力統計をプレイバックして表示する。プレイバック・データの省略時の表示時間は 3 秒間だが、スクリーンには監視された入出力統計が 5 秒間ずつ表示される。

2.

```
$ MONITOR/RECORD/NODISPLAY -
_ $ /BEGINNING=08:00:00 -
_ $ /ENDING=16:00:00 -
_ $ /INTERVAL=120 DISK
```

\$ MONITOR/INPUT/DISPLAY=HOURLY.LOG/INTERVAL=3600 DISK

このコマンドの列は、比較的短い間隔でデータを記録し、比較的長い間隔でデータをプレイバックしている。この方法は、さまざまな時間での平均値、最小値、最大値を求めるので、長い間隔で収集したときより正確な値が必要なときに便利である。

最初のコマンドは、指定された 8 時間に 2 分間隔で、システム上のすべてのディスクの入出力動作に関するデータを記録する。2 番目のコマンドは、1 時間ごとにデータをプレイバックして表示し、その内容を HOURLY.LOG というファイルに保存する。このファイルを表示またはプリントすれば、データを収集した 8 時間の 1 時間ごとのディスクの累計使用量が分かる。



注意:

HOURLY.LOG 内の現在の統計値は、新しい 3600 秒の収集期間ではなく、もとの 120 秒の収集期間における現在のデータを示しています。

3.

\$ MONITOR/INPUT/NODISPLAY/SUMMARY=DAILY.LOG DISK

前の例で作成された記録ファイルを使用し、データを収集した 8 時間の平均値を示す 1 ページの要約レポート・ファイルを作成する。要約レポートの形式は画面表示と同じになる。次に例を示す。

```

OpenVMS Monitor Utility
DISK I/O STATISTICS
  on node TLC
SUMMARY
From: 25-JAN-2002 08:00:00
To:   25-JAN-2002 16:00:00

```

I/O Operation Rate		CUR	AVE	MIN	MAX
DSA0:	SYSTEM_0	0.53	1.50	0.40	3.88
DSA1:	SYSTEM_1	0.00	0.39	0.00	8.38
DSA4:	WORK_0	0.00	0.11	0.00	1.29
DSA5:	WORK_1	0.03	0.87	0.00	5.95
DSA6:	WORK_2	0.03	0.25	0.00	2.69
DSA7:	WORK_3	0.04	0.97	0.00	20.33
DSA17:	TOM_DISK	0.00	0.04	0.00	0.80
DSA23:	MKC	0.00	0.00	0.00	0.13
\$4\$DUA0:	(RABBIT) SYSTEM_0	0.20	0.65	0.17	1.97
\$4\$DUA2:	(RABBIT) SYSTEM_0	0.20	0.65	0.17	1.97
\$4\$DUA3:	(RABBIT) SYSTEM_1	0.00	0.14	0.00	2.49

```

PLAYBACK
SUMMARIZING

```

6.7.7 記録した動作の遠隔プレイバック

適切な特権を持っていれば、DECnet によってローカル・システムに接続されている任意のシステムから MONITOR データを収集することができます。収集中のデータは、同時にローカル・システム上に表示することができます。その場合、次の手順に従います。

1. 各遠隔システム上の省略時の DECNET ディレクトリに MONITOR.COM という名前で次のような内容のファイルを作成する。

```
$ !
$ !      * Enable MONITOR remote playback *
$ !
$ MONITOR /NODISPLAY/RECORD=SYS$NET ALL_CLASSES
```

2. ローカル・システム上でデータを収集したい遠隔システムの論理名を定義する。
DEFINE 遠隔ノード論理名 ノード名::タスク = モニタ
ログイン・コマンド・プロシージャの中でアクセスしたいすべてのシステムの論理名を定義することもできる。
3. 遠隔システムからの MONITOR データを収集と同時に表示するためには、次の構文のコマンドを入力する。
MONITOR/INPUT= 遠隔ノード論理名 クラス名

MONITOR.COM ファイルを省略時の DECNET ディレクトリ以外のディレクトリに置き、アクセス制御文字列または代理のアカウントを使用して、これらのコマンド・ファイルを遠隔呼び出しすることも可能です。

MONITOR をローカル・システムで呼び出した場合は、遠隔システム上にコマンド・ファイル MONITOR.COM を実行するプロセスが生成されます。したがって、遠隔システムではこのプロセスに関連する CPU および DECnet のオーバーヘッドが発生します。このオーバーヘッドは、MONITOR.COM ファイルの中に /INTERVAL 修飾子とクラス名のリストを加えることによって制限できます。

混在バージョンのクラスタ・システムにおいて、遠隔で監視を行う方法については6.7.10 項「遠隔監視」で説明しています。

6.7.8 記録ファイルの更新

記録ファイルの更新は、プレイバックと記録の 2 つの操作を組み合わせて行われます。この機能を使用すると、記録ファイルのデータの量を減らすことができます。既存の記録ファイルをプレイバックする場合は、MONITOR のすべてのオプションを利用できます。したがって、記録されているデータから特定のクラス、時間帯、記録する間隔を選択することができます。

これらの操作により、記録されたデータの一部を削除した、サイズが小さな記録ファイルが新しく作成されます。収集する間隔を長くするとデータの量は少なくなり、それだけ新しい記録ファイルから表示または要約されるデータの精度は低くなります。この場合、平均の割合を示す値は影響されませんが、サンプル・データのサイズが小さいために、平均のレベルを示す値の精度は低くなります。次の例は、記録ファイルの更新方法を示しています。

例

```
$ SUBMIT MONREC.COM
```

MONREC.COM は次のコマンドを含んでいます。

```
$ MONITOR/NODISPLAY/RECORD/INTERVAL=60 /BEGINNING=8:00/ENDING=16:00 DECNET, LOCK
$ MONITOR/INPUT/NODISPLAY/RECORD DECNET
```

最初のコマンドはバッチ形式で動作し、午前 8 時から午後 4 時までの間、1 分ごとに DECnet とロック管理に関する情報を記録します。2 番目のコマンドは最初のコマンドが完了すると発行され、MONITOR.DAT ファイルの新しいバージョンを作成して DECnet に関するデータだけを再記録します。

6.7.9 MONITOR の継続実行

MONITOR をバックグラウンド・プロセスとして継続して実行することにより、システムの性能に関する情報を記録したデータベースを構築することができます。ここでは、クラスタ管理者として、マルチファイルのクラスタ全体の要約を作成するために使用できるプロシージャの例を示します。

このコマンド・プロシージャを自分のサイトに合うように、変更することができます。なお、SYSTARTUP.COM に論理名 SYS\$MONITOR および MON\$ARCHIVE を定義してからでないと、コマンド・ファイルを実行することはできません。

論理名 SYS\$EXAMPLES が指すディレクトリに、データベースの構築に利用できる 3 つのコマンド・プロシージャが含まれています。これらのプロシージャのインストールおよび実行の方法は、各プロシージャの先頭のコメントに示されています。表 6-10 「MONITOR のためのコマンド・プロシージャ」で、これらのプロシージャについて簡単にまとめます。

表 6-10 MONITOR のためのコマンド・プロシージャ

プロシージャ名	説明
MONITOR.COM	前回のブート時に作成された記録ファイルから要約ファイルを作成し、今回のブートの記録を開始する。記録の間隔は 10 分。
MONSUM.COM	複数のファイルから構成されるクラスタ全体の要約レポートを 2 種類作成して、システム管理者にメールする。一方のレポートには過去 24 時間の情報が記録され、もう一方のレポートには前日のプライム・タイム (午前 9 時から午後 6 時まで) の情報が記録される。このプロシージャは、毎日夜中に実行するように、自身をキューに再登録する。
SUBMON.COM	MONITOR.COM を独立プロセスとして実行する。サイト別スタートアップ・コマンド・プロシージャから SUBMON.COM を起動する。

MONITOR で継続的にデータを記録しながら、特定の期間の要約レポートを作成することができます。MONSUM.COM コマンド・プロシージャは毎日夜中に実行され、表 6-10 「MONITOR のためのコマンド・プロシージャ」に示す複数のファイルから構成される 2 つの要約レポートを生成し、メールします。これらのレポートはファイルに保存されません。内容を残すためには、メール・ファイルから情報を抽出するか、レポートを保存するように MONSUM.COM コマンド・プロシージャを変更します。

6.7.9.1 MONITOR.COM プロシージャの使用法

前回のブートで収集したデータから記録ファイルおよび要約ファイルを保存し、現在のブートのデータの連続記録を開始します。例 6-3 「MONITOR.COM プロシージャ」のプロシージャは、記録ファイルをパージしない点に注意してください。

例 6-3 MONITOR.COM プロシージャ

```
$ SET VERIFY
$ !
$ ! MONITOR.COM
$ !
$ ! This command file is to be placed in a cluster-accessible directory
$ ! called SYS$MONITOR and submitted at system startup time as a detached
$ ! process via SUBMON.COM. For each node, MONITOR.COM creates, in
$ ! SYS$MONITOR, a MONITOR recording file that is updated throughout the
$ ! life of the boot. It also creates, in MON$ARCHIVE, a summary file from
$ ! the recording file of the previous boot, along with a copy of that
$ ! recording file. Include logical name definitions for both cluster-
$ ! accessible directories, SYS$MONITOR and MON$ARCHIVE, in SYSTARTUP.COM.

$ !
$ SET DEF SYS$MONITOR
$ SET NOON
$ PURGE MONITOR.LOG/KEEP:2

$ !
$ ! Compute executing node name and recording and summary file names
$ ! (incorporating node name and date).

$ !
$ NODE = F$GETSYI("NODENAME")
$ SEP = ""
$ IF NODE .NES. "" THEN SEP = "_"
$ DAY = F$EXTRACT(0,2,F$TIME())
$ IF F$EXTRACT(0,1,DAY) .EQS. " " THEN DAY = F$EXTRACT(1,1,DAY)
$ MONTH = F$EXTRACT(3,3,F$TIME())

$ ARCHFILNAM = "MON$ARCHIVE:"+NODE+SEP+"MON"+DAY+MONTH
$ RECFIL = NODE+SEP+"MON.DAT"
$ SUMFIL = ARCHFILNAM+".SUM"
$ !
$ ! Check for existence of recording file from previous boot and skip
$ ! summary if not present.

$ !
$ OPEN/READ/ERROR=NORECFIL RECORDING 'RECFIL'
$ CLOSE RECORDING
$ !

$ !
$ ! Generate summary file from previous boot.
$ !
$ MONITOR /INPUT='RECFIL' /NODISPLAY /SUMMARY='SUMFIL' -
$ ALL_CLASSES+MODE/ALL+STATES/ALL+SCS/ITEM=ALL+SYSTEM/ALL+DISK/ITEM=ALL
$ !

$ !
$ ! Compute subject string and mail summary file to cluster manager.

$ !
$ !
$ A=""
$ B=" MONITOR Summary "
$ SUB = A+NODE+B+F$TIME()+A
$ MAIL/SUBJECT='SUB' 'SUMFIL' CLUSTER_MANAGER
$ !

$ !
$ ! Archive recording file and delete it from SYS$MONITOR.
```

```

$ !
$ COPY 'RECFIL' 'ARCHFILNAM'.DAT
$ DELETE 'RECFIL';*
$ !
$ NORECFIL:
$ SET PROCESS/PRIORITY=15
$ !

$ !
$ ! Begin recording for this boot. The specified /INTERVAL value is
$ ! adequate for long-term summaries; you might need a smaller value
$ ! to get reasonable "semi-live" playback summaries (at the expense
$ ! of more disk space for the recording file).

$ !
$ MONITOR /INTERVAL=300 /NODISPLAY /RECORD='RECFIL' ALL_CLASSES
$ !
$ !
$ ! End of MONITOR.COM
$ !

```

6.7.9.2 SUBMON.COM プロシージャの使用法

例 6-4 「SUBMON.COM プロシージャ」のプロシージャは、SYSTARTUP.COM から独立プロセスとして MONITOR.COM をキューに登録し、現在のブートの継続記録を開始します。

例 6-4 SUBMON.COM プロシージャ

```

$ SET VERIFY
$ !
$ ! SUBMON.COM

$ !
$ ! This command file is to be placed in a cluster-accessible directory
$ ! called SYS$MONITOR. At system startup time, for each node, it is
$ ! executed by SYSTARTUP.COM, following logical name definitions for
$ ! the cluster-accessible directories SYS$MONITOR and MON$ARCHIVE.
$ !

$ !
$ ! Submit detached MONITOR process to do continuous recording.
$ !

$ !
$ RUN SYS$SYSTEM:LOGINOUT.EXE -
      /UIC=[1,4] -
      /INPUT=SYS$MONITOR:MONITOR.COM -
      /OUTPUT=SYS$MONITOR:MONITOR.LOG -
      /ERROR=SYS$MONITOR:MONITOR.LOG -
      /PROCESS_NAME="Monitor" -

      /WORKING_SET=512 -
      /MAXIMUM_WORKING_SET=512 -
      /EXTENT=512/NOSWAPPING

$ !
$ !
$ ! End of SUBMON.COM
$ !

```

6.7.9.3 MONSUM.COM プロシージャの使用法

例 6-5 「MONSUM.COM プロシージャ」のプロシージャは、毎日のプライム・タイムのクラスタの要約を作成します。

例 6-5 MONSUM.COM プロシージャ

```
$ SET VERIFY
$ !
$ !   MONSUM.COM
$ !
$ !   This command file is to be placed in a cluster-accessible directory
$ !   called SYS$MONITOR and executed at the convenience of the cluster
$ !   manager.  The file generates both 24-hour and "prime time" cluster
$ !   summaries and resubmits itself to run each day at midnight.
$ !
$ SET DEF SYS$MONITOR
$ SET NOON

$ !
$ !   Compute file specification for MONSUM.COM and resubmit the file.
$ !
$ FILE = F$ENVIRONMENT("PROCEDURE")
$ FILE = F$PARSE(FILE,,, "DEVICE")+F$PARSE(FILE,,, "DIRECTORY")+F$PARSE(FILE,,, "NAME")
$ SUBMIT 'FILE' /AFTER=TOMORROW /NOPRINT

$ !
$ !   Generate 24-hour cluster summary.
$ !
$ !
$ !
$ MONITOR/INPUT=(SYS$MONITOR:*MON*.DAT;* ,MON$ARCHIVE:*MON*.DAT;*) -
/NODISPLAY/SUMMARY=MONSUM.SUM -
ALL_CLASSES+DISK/ITEM=ALL+SCS/ITEM=ALL-
/BEGIN="YESTERDAY+0:0:0.00" /END="TODAY+0:0:0.00" /BY_NODE

$ !
$ !
$ !   Mail 24-hour summary file to cluster manager and delete the file from
$ !   SYS$MONITOR.
$ !
$ !
$ MAIL/SUBJECT="Daily Monitor Clusterwide Summary" MONSUM.SUM CLUSTER_MANAGER
$ DELETE MONSUM.SUM;*

$ !
$ !   Generate prime-time cluster summary.
$ !
$ !
$ !
$ MONITOR/INPUT=(SYS$MONITOR:*MON*.DAT;* ,MON$ARCHIVE:*MON*.DAT;*) -
/NODISPLAY/SUMMARY=MONSUM.SUM -
ALL_CLASSES+DISK/ITEM=ALL+SCS/ITEM=ALL-
/BEGIN="YESTERDAY+9:0:0.00" /END="YESTERDAY+18:0:0.00" /BY_NODE

$ !
$ !
$ !   Mail prime-time summary file to cluster manager and delete the file
$ !   from SYS$MONITOR.
$ !
$ !
$ !
$ MAIL/SUBJECT="Prime-Time Monitor Clusterwide Summary" MONSUM.SUM CLUSTER_MANAGER
$ DELETE MONSUM.SUM;*
$ !
$ !   End of MONSUM.COM
$ !
```

このプロシージャの中の MAIL コマンドは、ファイルを CLUSTER_MANAGER というユーザに送信するように指定しています。CLUSTER_MANAGER のところを実際のユーザ名または論理名に置き換えてください。

多くの場合、データは多量になるため、要約ファイルはできるだけプリントするようにします。

6.7.10 遠隔監視

MONITOR は、転送メカニズムとして TCP/IP と DECnet の両方を使用できます。OpenVMS V7.0 以降では、TCP/IP を使用するためには、SYS\$STARTUP:SYSTARTUP_VMS.COM ファイルの中で次のコマンドを実行して、TCP/IP サーバを起動しておく必要があります。

```
$ @SYS$STARTUP:VPM$STARTUP.COM
```

DECnet は、ずっと作動し続けます。ネットワーク・オブジェクトは、要求の時に作成されません。

混合バージョンの OpenVMS Cluster システムにおける遠隔監視

MONITOR CLUSTER コマンドを発行する、または会話型の任意の MONITOR 要求で/NODE 修飾子を指定すると、OpenVMS Cluster システム内の任意のノードを開始することができます。

OpenVMS Cluster システムでの遠隔監視は、OpenVMS のバージョンが異なるノード間では、互換性がないことがあります。表 6-11 「OpenVMS Cluster システムでの遠隔の監視互換性」に、遠隔監視のバージョンの互換性を示します。

表 6-11 OpenVMS Cluster システムでの遠隔の監視互換性

バージョン	OpenVMS Alpha, 164 および VAX バージョン 6.n または 7.n	OpenVMS Alpha バージョン 1.5 および VAX バージョン 5.n
OpenVMS Alpha, 164, および VAX バージョン 6.n または 7.n	Yes	No
OpenVMS Alpha バージョン 1.5 および VAX バージョン 5.n	No	Yes

互換性のない遠隔ノードを監視しようとする、次のメッセージが表示されます。

```
%MONITOR-E-SRVMISMATCH, MONITOR server on remote node is an incompatible version
```

このメッセージが表示されたら、弊社のサポート担当者に連絡し、この問題を解決するための修正キットを入手してください。修正キットをインストールする前でも、MONITOR を使って、遠隔ノードについてのデータを得ることができます。これを行うには、遠隔ノードについてのデータを記録してから MONITOR プレイバック機能を実行し、ローカル・ノードについてのデータを検査します。

OpenVMS Cluster システムにおいて遠隔ノードを監視する際には、もう 1 つの相違があります。OpenVMS バージョン 6.2 以降では、監視できるディスク数の制限が、レコードの出力については 799 から 909 に、表示と要約の出力については 799 から 1817 に増えました。ただし、OpenVMS バージョン 6.2 以降を実行している遠隔ノードを、OpenVMS バージョン 6.2 より前のバージョンが実行されているシステムで監視する場合は、制限値は 799 のままです。

MONITOR についての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

第7章 資源使用状況の調査

この章では、システム資源がどのように使用されているかを調べる方法を説明します。この情報は、次のことに役立てることができます。

- 各ユーザに対するシステム使用料の請求。個々のユーザが使用した資源のレポートを作成することができる。
- 将来必要となる機器の検討。資源の使用パターンの変化が分かり、将来どの資源が必要になるかを予測することができる。
- システムのトラブルシューティング。各プロセスが最後に終了したときの終了状態をチェックすることができる。
- システムの性能の改善。個々のイメージおよびプロセスがシステムにどのくらいの負荷を与えているか調べることができる。
- 機密保護を侵害するような動きの検出。資源の使用パターンがどのようなときに通常でないかを識別することができる。

この章の内容

この章では、次の作業を説明します。

作業	参照箇所
調査対象の資源の表示	7.2 項 「調査対象の資源の決定」
調査対象の資源の制御	7.3 項 「調査対象の資源の制御」
会計情報ファイルの更新	7.4 項 「会計情報ファイルの更新」
会計情報ファイルの移動	7.5 項 「会計情報ファイルの移動」
資源使用量を示すレポートの生成	7.6 項 「資源の使用量を示すレポートの作成」
会計グループの設定	7.7 項 「会計グループの設定」
ディスク空間使用量の調査	7.8 項 「ディスク空間使用量の調査」

さらに、次の項目について説明します。

項目	参照箇所
会計情報ファイル	7.1 項 「会計情報ファイルについて」

7.1 会計情報ファイルについて

システムは、特定の資源の使用量に関する情報を収集し、**会計情報ファイル**と呼ばれるファイルに記録します。このような情報の代表的なものに、各プリント・ジョブによって使用される CPU 時間があります。

省略時の設定において、情報収集の対象となる資源は、使用するコンピュータのモデルによって異なります。システム管理者は、どの資源の情報を収集するか制御することができます。たとえば、資源の使用量を調査する必要がない場合は、会計情報ファイルへの記録を停止することもできます (7.3 項 「調査対象の資源の制御」を参照)。

会計情報ファイルは OpenVMS Cluster 内の各ノードに存在します。各ノードの情報を記録しているファイルを、そのノードの**現在の会計情報ファイル**といいます。省略時の会計情報ファイルは SYS\$MANAGER:ACCOUNTING.DAT ですが、必要に応じて変更することもできます (7.5 項 「会計情報ファイルの移動」を参照)。

会計情報ファイルには情報がバイナリ形式で記録されます。したがって、TYPE コマンドでこのファイルの内容を表示することはできません。内容を表示する場合は、ACCOUNTING ユーティリティを使用します (7.6 項 「資源の使用量を示すレポートの作成」を参照)。

7.2 調査対象の資源の決定

現在、情報を収集している資源を決定する場合には、SHOW ACCOUNTING コマンドを使用します。

\$ SHOW ACCOUNTING

このコマンドは、次の 2 つのカテゴリについて、そのキーワードを表示します。

- 調査対象となっている資源の種類を示すキーワード。

キーワード	資源の種類
IMAGE	イメージによって使用される資源
LOGIN_FAILURE	ログインに失敗した場合に使用される資源
MESSAGE	\$\$NDJBC システム・サービスの呼び出しによって会計情報ファイルに記録される書式化されていない資源レコード
PRINT	プリント・ジョブによって使用される資源
PROCESS	プロセスによって使用される資源

- 調査対象となっているプロセスの種類を示すキーワード。プロセスあるいはイメージのための資源が記録されている場合、これらのキーワードはプロセスの種類を示す。

キーワード	プロセス
BATCH	バッチ・プロセス
DETACHED	独立プロセス
INTERACTIVE	会話型プロセス
NETWORK	ネットワーク・プロセス
SUBPROCESS	サブプロセス (親プロセスはバッチ、独立、会話型、ネットワークのいずれの場合もある。)

例

\$ SHOW ACCOUNTING

Accounting is currently enabled to log the following activities:

```
PROCESS      any process termination
IMAGE        image execution
INTERACTIVE  interactive job termination
LOGIN_FAILURE login failures
NETWORK      network job termination
PRINT        all print jobs
```

この例のキーワードから、ローカル・ノードでは次の資源が調査対象になっていることが分かります。

- 会話型プロセスおよびネットワーク・プロセスによって使用される資源。
- 会話型プロセスおよびネットワーク・プロセスで動作しているイメージによって使用される資源。

- 正常に行われないログインによって使用される資源。
- プリント・ジョブによって使用される資源。

7.3 調査対象の資源の制御

システム管理者は、会計情報ファイルにどの資源の情報を記録するかを制御することができます。調べる必要がない資源の記録を停止すれば、ディスク空間の使用量を減らすこともできます。

作業方法

1. 調査対象にする資源を一時的に制御する場合は、SET ACCOUNTING コマンドで修飾子 /ENABLE または /DISABLE を次の形式で使用する。
SET ACCOUNTING/DISABLE[=(キーワード [,...])] /ENABLE[=(キーワード [,...])]
キーワードは、7.2 項「調査対象の資源の決定」で説明したものと同一。
2. 調査対象にする資源を変更したい場合 (すなわち、リブート後も同じ設定を使用したい場合) は、スタートアップ・ファイル SYS\$MANAGER:SYSTART_VMS.COM にある SET ACCOUNTING の記述を変更する。

例

会話型プロセスおよびバッチ・プロセスによって使用されるものを除くすべての資源の記録を停止する場合は、次のコマンドを使用します。

```
SET ACCOUNTING/DISABLE/ENABLE=(PROCESS,INTERACTIVE,BATCH)
```

/DISABLE 修飾子の後ろにはキーワードが付いていません。したがって、この修飾子によりすべての資源の記録が停止されます。次の /ENABLE 修飾子により、会話型プロセスおよびバッチ・プロセスによって使用される資源の記録が開始されます。

7.4 会計情報ファイルの更新

会計情報ファイルを更新する場合は、次のコマンドを使用します。

```
SET ACCOUNTING/NEW_FILE
```

このコマンドにより、現在の会計情報ファイルがクローズし、新しいバージョンがオープンします。

現在の会計情報ファイルに書き込みを行おうとするときにエラーが検出されると、会計情報ファイルは自動的にクローズし、新しいバージョンがオープンします。

例

現在の会計情報ファイルをクローズして新しいバージョンをオープンし、さらに古いファイルの名前を WEEK_24_RESOURCES.DAT に変更します。古いファイルはその週に使用された資源の記録として使用することができます。

```
SET ACCOUNTING/NEW_FILE
```

```
RENAME SYS$MANAGER:ACCOUNTNG.DAT;-1 WEEK_24_RESOURCES.DAT
```

7.5 会計情報ファイルの移動

システムを初めてインストールした場合、現在の会計情報ファイルは SYS\$MANAGER:ACCOUNTNG.DAT となります。

このファイルは非常に大きくなるため、システムの性能が低下しないように、適切な時期にシステム・ディスクから移動することが大切です。

作業方法

1. システム論理名テーブルにある ACCOUNTNG という論理名で使用するファイルを定義する。たとえば、次のようにする。

```
$ DEFINE ACCOUNTNG MYDISK:MYFILE.DAT/SYSTEM
```

デバイスおよびディレクトリを含む完全ファイル指定を使用する。



注意:

2 つのノードの情報を 1 つの会計情報ファイルに情報を記録することはできません。それぞれのノードで定義した ACCOUNTING が同一のファイルを指している場合、各ノードごとに独自のバージョンがオープンします。

2. リブート後も同じ定義を継続したい場合は、この定義を SYS\$MANAGER:SYLOGICALS.COM ファイルに追加する。
3. SET ACCOUNTING コマンドに /NEW_FILE を指定して、新しいファイルを準備する。

```
$ SET ACCOUNTING/NEW_FILE
```

例

現在の会計情報ファイルを MYDISK:MYFILE.DAT に変更する場合は、次のようにします。

```
$ DEFINE ACCOUNTNG MYDISK:MYFILE.DAT/SYSTEM
```

```
$ SET ACCOUNTING/NEW_FILE
```

7.6 資源の使用量を示すレポートの作成

レポートには次の 3 種類があります。

レポートの種類	使用する修飾子
簡略レポート	/BRIEF (省略時の設定)
詳細レポート	/FULL
要約レポート	/SUMMARY

レポートを作成する場合、ACCOUNTING コマンドに該当する修飾子を指定します。

ACCOUNTING [ファイル指定 [...]/修飾子 [...]]

このコマンドにより、ACCOUNTING ユーティリティが実行されます。パラメータの **ファイル指定**には、処理したい会計情報ファイルを指定します。複数のファイルを指定する場合は、それぞれをコンマで区切ります。このパラメータを省略すると、省略時の現在の会計情報ファイル SYS\$MANAGER:ACCOUNTNG.DAT が処理されます。

省略時の設定では、指定した会計情報ファイルの中のすべてのレコードが処理されますが、修飾子を使用して処理対象のレコードを指定することもできます。

省略時の設定では、簡略レポートおよび詳細レポートのレコードの順序は、会計情報ファイルに記録された順序になります。簡略レポートおよび省略レポートでは、レコードの順序を変更することもできます。その場合は、/SORT 修飾子を使用します。

例

論理名 ACCOUNTNG によって指定されるファイル内の情報の簡略レポートを生成する例です。/TYPE 修飾子によって、処理対象のレコードをプリント・ジョブに関するものに限定しています。また、/SORT 修飾子は、ユーザ名のアルファベットの逆の順序に従って各レコードを並べます。

```
$ ACCOUNTING ACCOUNTNG/TYP=PRINT/SORT=USER
```

Date / Time	Type	Subtype	Username	ID	Source	Status
13-APR-2000 13:36:04	PRINT		SYSTEM	20A00442		00000001
13-APR-2000 12:42:37	PRINT		JONES	20A00443		00000001
13-APR-2000 14:43:56	PRINT		FISH	20A00456		00000001
14-APR-2000 19:39:01	PRINT		FISH	20A00265		00000001
14-APR-2000 20:09:03	PRINT		EDWARDS	20A00127		00000001
14-APR-2000 20:34:45	PRINT		DARNELL	20A00121		00000001
14-APR-2000 11:23:34	PRINT		CLARK	20A0032E		00040001
14-APR-2000 16:43:16	PRINT		BIRD	20A00070		00040001
14-APR-2000 09:30:21	PRINT		ANDERS	20A00530		00040001

7.7 会計グループの設定

各ユーザは、登録時に UIC セキュリティ・グループのいずれかに所属するようになっています。多くの場合、セキュリティ・グループを会計の目的で使用するのには適切ではありません。ACCOUNTING ユーティリティの /ACCOUNT 修飾子を使用すれば、システムに登録されているユーザをいくつかの会計グループに分けることができます。

ACCOUNTING ユーティリティを使用すれば、次のことができます。

- 特定の会計グループまたはセキュリティ・グループに属するすべてのユーザの資源使用量をまとめる。会計グループによってまとめるときは /ACCOUNT 修飾子、セキュリティ・グループによってまとめるときは /UIC 修飾子を使用する。
- 特定の会計グループまたはセキュリティ・グループに属するすべてのユーザのレコードを選択する。会計グループによって選択するときは /ACCOUNT 修飾子、セキュリティ・グループによって選択するときは /UIC 修飾子を使用する。

作業方法

1. 会計グループの構成を考える。どのような会計グループを使用するか、各ユーザをどの会計グループに所属させるか、各グループの名前をどのようにするかを決める。

会計グループの名前は最大 8 文字。

2. UAF の会計情報フィールドの値を変更する。AUTHORIZE ユーティリティの MODIFY コマンドを使用して、各ユーザの会計情報フィールドの値をそのユーザが所属する会計グループの名前に変更する。

MODIFY ユーザ名/ACCOUNT= 会計グループ名

ユーザ名	ユーザの名前
会計グループ名	そのユーザを所属させたい会計グループの名前

該当する会計情報フィールドが変更されたユーザは、次のログインから新しい会計グループに所属します。そのユーザが使用した資源が記録される際には、該当する会計グループ名が同時に記録されるようになります。

例

FORD というユーザの会計グループ名を SALES_W8 に変更する場合は、次のコマンドを実行します。

```
$ RUN SYS$SYSTEM:AUTHORIZE
UAF> MODIFY FORD/ACCOUNT=SALES_W8
UAF> EXIT
```

7.8 ディスク空間使用量の調査

あるユーザが使用しているディスク空間の量を調べる場合は、SYSMAN コマンドを使用します。また、ディスク・クォータを有効にしていない場合には、DIRECTORY コマンドを使用します。

作業方法

次のいずれかの方法を使用します。

- 次の形式で、SYSMAN の DISKQUOTA SHOW コマンドを使用する。
DISKQUOTA SHOW 所有者 [/DEVICE= デバイス指定]
指定したユーザが指定したディスク上で所有している全ファイルの使用ブロック数が示される。
- 次の形式で、DIRECTORY コマンドに /SIZE 修飾子と /GRAND_TOTAL 修飾子を指定する。
DIRECTORY [ファイル指定 [...]]/SIZE=ALLOCATION/GRAND_TOTAL
指定したディレクトリにある全ファイルにより使用されているブロック数が示される。
なお、DIRECTORY コマンドには、ファイルのヘッダやユーザのルート・ディレクトリで使用されているブロックは含まれない。

例

1. 次の例では、SYSMAN を使用して各ユーザが所有している全ファイルの使用ブロック数を調べる。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> DISKQUOTA SHOW *

%SYSMAN-I-QUOTA, disk quota statistics on device SYS$SYSTEM:MYDISK
Node UNION
      UIC                Usage      Permanent Quota  Overdraft Limit
[0,0]                   0          1000             100
[DOC,EDWARDS]          115354      150000           5000
[DOC,FISH]             177988      250000           5000
[DOC,SMITH]            140051      175000           5000
[DOC,JONES]            263056      300000           5000
```

2. 次の例では、DIRECTORY コマンドを使用して、MYDISK:[PARSONS] およびその下のディレクトリにある全ファイルの使用ブロック数を表示する。

```
$ DIRECTORY MYDISK:[PARSONS...]/SIZE=ALLOCATION/GRAND_TOTAL

Grand total of 28 directories, 2546 files, 113565 blocks.
```

第8章 OpenVMS Cluster の管理

この章では、OpenVMS Cluster 環境に関連する概念について説明します。また、SHOW CLUSTER ユーティリティを使用してクラスタに関する情報を表示する方法、および SYSMAN ユーティリティを使用して OpenVMS Cluster 環境を管理する方法を説明します。

この章の内容

この章では次の作業を説明します。

作業	参照箇所
SHOW CLUSTER コマンドの使用の開始	8.3.2 項 「SHOW CLUSTER コマンドの起動」
レポートへの情報の追加	8.3.3 項 「レポートへの情報の追加」
データの表示の制御	8.3.4 項 「表示データの制御」
データの表示の書式化	8.3.5 項 「表示データの書式化」
スタートアップ初期化ファイルの作成	8.3.6 項 「SHOW CLUSTER スタートアップ初期化ファイルの作成」
SHOW CLUSTER コマンドを含むコマンド・プロシージャの使用	8.3.7 項 「SHOW CLUSTER ユーティリティのコマンドを含むコマンド・プロシージャの使用」
SYSMAN による機密保護の管理	8.5 項 「システム管理ユーティリティ (SYSMAN) による機密保護の管理」
SYSMAN DO コマンドによる OpenVMS Cluster の管理	8.6 項 「SYSMAN DO コマンドによる OpenVMS Cluster の管理」

さらに、次の項目について説明します。

項目	参照箇所
OpenVMS Cluster システムについて	8.1 項 「OpenVMS Cluster システムについて」
OpenVMS Cluster 環境の設定	8.1.1 項 「OpenVMS Cluster 環境の設定」
クラスタ全体のシステム管理	8.1.2 項 「クラスタ全体のシステム管理」
SHOW CLUSTER ユーティリティ	8.3.1 項 「SHOW CLUSTER ユーティリティについて」
SYSMAN および OpenVMS Cluster 管理について	8.4 項 「SYSMAN ユーティリティと OpenVMS Cluster 管理について」

8.1 OpenVMS Cluster システムについて

OpenVMS Cluster システムとは、2 台以上のコンピュータ、および記憶サブシステムを任意に組み合わせたシステム構成です。OpenVMS Cluster は、下記のシステム 2 台以上から構成されます。

- VAX システム
- Alpha システム
- I64 システム
- VAX システムと Alpha システム
- Alpha システムと I64 システム

OpenVMS Cluster システムでは、システム資源のすべてまたは一部を共有しますが、ユーザからはクラスタ全体が単一のシステムに見えます。コンピュータのグループがクラスタ全体の資源を共有すると、すべてのコンピュータが持つ記憶資源およびシステム資源が結合され、システムの処理能力、通信能力、および可用性を向上させることができます。

OpenVMS Cluster システムのどのノードからもアクセスあるいは使用できる資源 (ディスクやキューなど) を**共有資源**と呼びます。データ・ファイル、アプリケーション・プログラム、およびプリンタは、それが物理的に存在するノードに関係なく、共有資源としてクラスタ上のどのユーザからもアクセスできる数少ないアイテムです。

ディスクが OpenVMS Cluster 環境内で共有資源として設定されていると、各ユーザはどのノードからログインしても同じ環境 (パスワード、特権、省略時のログイン・ディスクへのアクセス権など) を持つこととなります。ディスクの共有により、大容量記憶デバイスをより効率的に使用することができます。これは、任意のデバイスの情報を複数のノードから利用できるので、必要とするそれぞれの場所に情報を書き込む必要がなくなるためです。 OpenVMS MSCP (大容量記憶制御プロトコル)、または TMSCP (テープ大容量記憶制御プロトコル) を使用すると、直接記憶デバイスが接続されていないノードでも、サーバ・ソフトウェアによりテープを利用できるようになります。

プリント・キューおよびバッチ・キューも共有資源として設定することができます。 OpenVMS Cluster システムでプリント・キューおよびバッチ・キューを共有する場合、すべてのノードで使用されるキューが 1 つのキュー・データベースによって管理されます。このキュー・データベースにより、どのノードからもキューを利用できます。たとえば、ALBANY、BASEL、および CAIRO という 3 つのノードを持つクラスタ・システムで、各ノードが持つ資源が完全に共有されていれば、ALBANY というノードにログインしたユーザは、BASEL に物理的に存在するファイルを、CAIRO に物理的に接続されているプリンタに送信できます。このとき、ユーザはこれらのファイルおよびプリンタが存在するノードを指定する必要はありません。また、各資源の存在場所を知る必要もありません。

OpenVMS Cluster システムの計画

OpenVMS Cluster ではさまざまな構成が可能です。サポートされているデバイスと構成についての詳細は、『OpenVMS Cluster 構成ガイド』およびソフトウェア仕様書 (SPD) を参照してください。

ここでは、OpenVMS Cluster システムについて簡単に説明します。 OpenVMS Cluster 環境の設定方法と使用方法についての詳細は、『OpenVMS Cluster システム』を参照してください。

8.1.1 OpenVMS Cluster 環境の設定

計画したシステム構成に基づいて必要なハードウェアをインストールし、各ハードウェアが適切に動作することが確認できたら、さまざまなシステム・ソフトウェア機能を使って OpenVMS Cluster システムを設定することができます。クラスタを構築するための作業は次のとおりです。

作業内容	参照するドキュメント
第 1 の OpenVMS Cluster コンピュータ上でのオペレーティング・システムのインストールまたはアップグレード	使用するコンピュータのインストール・ガイド およびオペレーション・ガイド
必要なソフトウェア・ライセンスのインストール	『OpenVMS License Management Utility Manual』
DECnet for OpenVMS ネットワークの構成と使用開始	『DECnet for OpenVMS Networking Manual』
TCP/IP Services の構成と使用開始	『TCP/IP Services for OpenVMS インストール/コンフィギュレーション』

作業内容	参照するドキュメント
クラスタ・オペレーティング環境を定義するファイルとディスク操作およびキュー動作を制御するファイルの準備	『OpenVMS Cluster システム』
クラスタへのコンピュータの追加	『OpenVMS Cluster システム』

これらの作業を行う順序は各作業環境のさまざまな要因、さらには同じ作業環境の各クラスタのさまざまな要因によって異なります。

8.1.2 クラスタ全体のシステム管理

システムがインストールされた後、システム管理者は、必要な機密保護を維持しながら最大の生産性と効率を得るために、ユーザおよび資源をどのように管理するかを検討する必要があります。OpenVMS Cluster システムは、環境の条件に合うようにユーザと資源を配分できる柔軟性を備えています。また、条件の変化に合わせて資源を容易に配分し直すことができます。OpenVMS Cluster システムで利用できる資源の数が膨大であっても、それらを1つのシステムとして管理することができます。

クラスタを統合して管理するためのツールや製品が用意されています。

OpenVMS Cluster 管理用ツール

オペレーティング・システムには次のシステム管理用ユーティリティが用意されています。

ユーティリティ	説明
HP Availability Manager	同時に複数のノードからデータを収集および解析し、すべての出力を集中化した Windows PC または DECwindows に表示する (8.2 項「HP OpenVMS Availability Manager によるデータの解析」および『HP OpenVMS Availability Manager User's Guide』を参照してください。)
Monitor ユーティリティ (MONITOR)	基本性能データを提供する (6.7 項「オペレーティング・システムの性能の監視」を参照)。
SHOW CLUSTER ユーティリティ	OpenVMS Cluster システム内の動作を監視し、その動作の情報を収集してターミナルなどの出力デバイスに送信する (8.3 項「SHOW CLUSTER ユーティリティの使用法」を参照)。
SYSMAN ユーティリティ	クラスタ内のノードの全部あるいは一部に共通の制御コマンドを送信する (8.6 項「SYSMAN DO コマンドによる OpenVMS Cluster の管理」を参照)。

システム管理用アプリケーション

次の製品を利用することができます。ただし、これらの製品は OpenVMS オペレーティング・システムに付属していません。

製品	説明
POLYCENTER Solutions	広範囲の操作管理製品。複雑な分散環境の管理を補助する。POLYCENTER Software Installation ユーティリティについては、『OpenVMS システム管理者マニュアル (上巻)』のインストールの章を参照。
Storage Library System (SLS) for VAX ¹ Archive Backup System (ABS) ²	テープ、カートリッジ・テープ、および光ディスクを使用できるようにするソフトウェア・ツール・セット。
OpenVMS Cluster Console System (VCS)	OpenVMS Cluster システムのコンソール管理を1つのコンソール・ターミナル上で統合して行うシステム。

¹ VAX のみ

² Alpha および I64 のみ

上記のシステム管理ツールについての詳細は、それぞれの製品に関するドキュメントを参照してください。

8.2 HP OpenVMS Availability Manager によるデータの解析

Availability Manager は、リアルタイムの監視、診断、修正を行うツールであり、システム管理者が OpenVMS システムや OpenVMS Cluster の可用性を改善するのを支援します。Availability Manager は、システム・プログラマやアナリストが、詳しい分析のために特定のノードやプロセスをターゲットにしたり、システム・オペレータやサービス技術者が、ハードウェアやソフトウェアの問題を解決するのに役立ちます。

Availability Manager は、複数のノードからシステム・データとプロセス・データを同時に収集して分析し、結果を Windows PC または DECwindows Motif ディスプレイに表示します。Availability Manager は、収集したデータに基づいて、資源と拒否の問題を修正する措置を、リアルタイムで分析して検出し、提案します。

詳細は『HP OpenVMS Availability Manager User's Guide』を参照してください。

8.3 SHOW CLUSTER ユーティリティの使用法

SHOW CLUSTER ユーティリティは、OpenVMS Cluster 内のノードを監視します。このユーティリティにより、クラスタの動作と性能に関する情報を表示することができます。

この節では、SHOW CLUSTER ユーティリティを取り上げ、次の作業方法を示します。

作業	参照箇所
SHOW CLUSTER ユーティリティの起動	8.3.2 項 「SHOW CLUSTER コマンドの起動」
レポートへの情報の追加	8.3.3 項 「レポートへの情報の追加」
表示データの制御	8.3.4 項 「表示データの制御」
データの形式と表示	8.3.5 項 「表示データの書式化」
スタートアップ初期化ファイルの作成	8.3.6 項 「SHOW CLUSTER スタートアップ初期化ファイルの作成」
SHOW CLUSTER ユーティリティのコマンドを含むコマンド・プロシージャの使用	8.3.7 項 「SHOW CLUSTER ユーティリティのコマンドを含むコマンド・プロシージャの使用」

8.3.1 SHOW CLUSTER ユーティリティについて

SHOW CLUSTER からの情報は、ターミナルの画面に表示したり、デバイスまたはファイルに送信したりできます。SHOW CLUSTER の使用方法としては、会話形式で実行する方法、コマンド・プロシージャに記述する方法、または省略時の設定を定義する初期化ファイルに記述する方法があります。このユーティリティは CMKRNL 特権でインストールされるので、SHOW CLUSTER は通常の特権で実行できます。

SHOW CLUSTER が収集する情報は、およそ 100 フィールドにも及びます。そこで、レポートの書式をカスタマイズしたり、出力内容を特に必要なデータに限定するようにレポートを定義することができます。

SHOW CLUSTER のレポートは、クラスとフィールドによって構成されます。

構成単位	説明
クラス	関連する1つ以上のフィールドを1つにまとめたもの。クラス名を使って、そのクラスに属するフィールド全体をまとめてレポートに追加したりレポートから削除したりできる。各クラスにはいくつかのフィールドが定義されている。また、いくつかのクラスでは、新しいフィールドの追加あるいは既存のフィールドの削除を行うことができる。
フィールド	レポートの各欄に対応する。複数のフィールドに対して同じ名前を付けることはできない。フィールド名を使って、SHOW CLUSTER レポートにフィールドを追加したり、レポートからフィールドを削除したりできる。 各クラスのすべてのフィールドの名前と内容については、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』の SHOW CLUSTER の章を参照。

省略時の SHOW CLUSTER レポートに、フィールドおよびクラスを追加することができます。連続表示の途中でフィールドまたはクラスを追加すると、追加したフィールドまたはクラス的数据は自動的に表示内容に追加されます。

図 8-1 「省略時の SHOW CLUSTER レポート」は、省略時の SHOW CLUSTER レポートの例です。省略時のレポートには、SYSTEMS および MEMBERS の2つのクラスの情報が含まれます。各クラス名の下には、各クラスの情報に対応するフィールドの欄が表示されます。

図 8-1 省略時の SHOW CLUSTER レポート

View of Cluster from system ID 65536 node: CLUB 31DEC1997 14:00:00

SYSTEMS		MEMBERS
NODE	SOFTWARE	STATUS
CLUB	VMS V7.2	MEMBER
HSJ400	HSJ V25J	
HSC900	HSC V860	
CHIP	VMS V7.1	MEMBER
DISK3	RFX V256	
DISK1	RFX V256	
SPREE	VMS V6.2	MEMBER
SPRITZ	VMS V7.1	MEMBER

ZK8998AG E

表 8-1 「省略時の SHOW CLUSTER レポートに含まれるフィールド」では、図 8-1 「省略時の SHOW CLUSTER レポート」の各フィールドについて簡単に説明します。

表 8-1 省略時の SHOW CLUSTER レポートに含まれるフィールド

フィールド	説明
NODE	遠隔システムのノード名。通常、ノード名は、クラスタの管理者が SYSGEN の SCSNODE パラメータによって、DECnet ノード名と同じ名前に設定する。
HW_TYPE	ハードウェアのタイプと遠隔システムのモデル。
SOFTWARE	遠隔システムで現在稼働しているオペレーティング・システムの名前とバージョン。
STATUS	クラスタ内のノードの状態。このフィールドの値が MEMBER のとき、そのシステムはクラスタの構成メンバである。

SHOW CLUSTER レポートを使用していくうちに、レポートの中でどのフィールドやクラスが重要であるかが分かってきます。そこで、スタートアップ初期化ファイルを作成して、省略時のレポート書式を定義することができます。また、SHOW CLUSTER を会話形式で実行しながらコマンド・プロシージャを作成することができます。このように、必要なデータを示すレポートを短い時間で定義することができます。なお、スタートアップ初期化ファイルとコマンド・プロシージャについては、この章で後ほど説明します。

SHOW CLUSTER は多くのフィールドを出力するため、レポートはすぐに画面に表示しきれなくなり、そこで、SHOW CLUSTER には次のような表示内容を制限する機能が備えられています。

- 38 の SHOW CLUSTER コマンド
- 省略時のキーパッド (再定義可能)

これらの機能についての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

8.3.2 SHOW CLUSTER コマンドの起動

SHOW CLUSTER ユーティリティを使用する場合は、SHOW CLUSTER コマンドを入力します。修飾子を何も指定しないで実行すると、図 8-1 「省略時の SHOW CLUSTER レポート」に示す省略時のレポートが表示された後、DCL プロンプトに戻ります。

一方、レポートを継続して表示しながら、途中で SHOW CLUSTER コマンドを入力して出力内容を変更することができます。たとえば、表示中のレポートに特定のクラスやフィールドの情報を追加したり、レポートから削除することができます。このような連続表示セッションを呼び出す場合は、SHOW CLUSTER コマンドに /CONTINUOUS 修飾子を指定して実行します。なお、SHOW CLUSTER コマンドで使用する修飾子については、8.3.2.3 項「SHOW CLUSTER での修飾子の使用」で説明します。

作業方法

省略時の SHOW CLUSTER レポートを連続モードで表示する場合は、次のコマンドを入力します。

```
§ SHOW CLUSTER/CONTINUOUS
```

SHOW CLUSTER は省略時のレポートを表示します。通常、表示内容は 15 秒ごとに更新され、前回から変更された部分は反転して表示されます。連続表示セッションを呼び出すと、省略時のレポートが表示された後で次のプロンプトが表示されます。

```
Command>
```

レポートの内容が多くて 1 つの画面に収まらないときは、Command> プロンプトが見えなくなります。その場合は Return キーを押すとプロンプトが表示されます。

次の項では、SHOW CLUSTER ユーティリティに関する次の作業の方法を説明します。

作業	参照箇所
画面に表示されていない情報の表示	8.3.2.1 項「画面に表示されていない情報の表示」
連続表示セッションの終了	8.3.2.2 項「連続表示セッションの終了」
SHOW CLUSTER での修飾子の使用	8.3.2.3 項「SHOW CLUSTER での修飾子の使用」

8.3.2.1 画面に表示されていない情報の表示

PAN コマンドを使用すると、レポート全体をカラム単位で左右に移動したり、行単位で上下に移動して、表示されていない部分を見ることができます。



注意:

PAN コマンドで画面の下で見えていない部分を表示させると、レポートのヘッダが画面からなくなり、ヘッダを画面に表示させたままレポートをスクロールするためには、SCROLL コマンドを使用します。画面上に複数のレポートを表示している時に SCROLL コマンドを使用する場合には、スクロール対象のレポートを選択するという作業も必要になります。SCROLL コマンドについては、8.3.5.4 項「レポートのスクロール」で説明します。

作業方法

画面に表示されていない部分を表示させるためには、次のいずれかの方法を用います。

- Command> プロンプトに対して PAN コマンドを入力する。たとえば、次のコマンド行を入力する。

```
Command> PAN DOWN 10
```

表示される部分が 10 行分下に移動する。

- 矢印キーを PAN コマンドとして定義する。

```
Command> SET FUNCTION PAN
```

このコマンドにより、各矢印キーが次のようなコマンドとして定義される。

矢印キー	対応するコマンド
↑	PAN UP 1
↓	PAN DOWN 1
→	PAN RIGHT 1
←	PAN LEFT 1

このような再定義を行うと、矢印キーを使ってレポートを上下左右に移動することができる。

詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』の SET FUNCTION コマンドおよび PAN コマンドの説明を参照。

矢印キーの再設定

省略時の設定では、SHOW CLUSTER の矢印キーは EDIT 機能に設定されます。したがって、コマンド・プロンプトでは、DCL の行モード編集に似たコマンド行編集を行うことができます。たとえば、左矢印キーを押すとカーソルは左に移動し、上矢印キーを押すと直前に実行したコマンドが呼び出されます。DCL の行モード編集については、『OpenVMS ユーザーズ・マニュアル』を参照してください。

SET FUNCTION コマンドを使用する場合は、ファンクション・キーを再設定してください。その後で矢印キーが再定義され、DCL の行モード編集が利用できなくなります。

矢印キーを再設定するには、次のコマンドを実行します。

```
Command> SET FUNCTION EDIT
```

8.3.2.2 連続表示セッションの終了

連続表示セッションを終了するためには、次のいずれかの方法を用います。

- 次のいずれかの方法で DCL プロンプトに戻る。
 - Command> プロンプトで EXIT と入力する。
 - Ctrl/Z を押す。

- Ctrl/Y を押す。
- Ctrl/C を押して、レポートの内容を表示したまま終了する。

8.3.2.3 SHOW CLUSTER での修飾子の使用

SHOW CLUSTER のコマンドで利用できる修飾子を表 8-2 「SHOW CLUSTER で利用できる修飾子」に示します。詳細は『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』を参照してください。

表 8-2 SHOW CLUSTER で利用できる修飾子

修飾子	機能
/BEGINNING= 時刻	SHOW CLUSTER セッションを開始する時刻を指定する。
/CONTINUOUS	SHOW CLUSTER の表示内容を継続的に更新する連続表示モードで実行するかどうかを指定する。
/ENDING= 時刻	SHOW CLUSTER セッションを終了する時刻を指定する。
/INTERVAL= 秒	レポートの内容を更新する間隔を秒単位で指定する。
/OUTPUT= ファイル指定	SY\$OUTPUT で指定されているデバイスの代わりに、出力するファイルを指定する。

例

連続表示モードでは、表示内容が 15 秒間隔で更新されるようにあらかじめ設定されています。この間隔は /INTERVAL 修飾子を使って変更することができます。

```
§ SHOW CLUSTER/CONTINUOUS/INTERVAL=5
```

このコマンドの実行により、レポートが 5 秒ごとに更新され、変更された内容は反転して表示されるようになります。

8.3.3 レポートへの情報の追加

SHOW CLUSTER によって表示されるレポートには、入手可能な情報の一部しか表示されません。図 8-1 「省略時の SHOW CLUSTER レポート」に示したように、省略時の設定で表示されるクラスは MEMBERS および SYSTEMS です。表 8-3 「SHOW CLUSTER レポートで入手可能なクラス情報」では、SHOW CLUSTER レポートに表示できるすべてのクラスを簡単に説明します。これらのクラスについての詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』を参照してください。

表 8-3 SHOW CLUSTER レポートで入手可能なクラス情報

クラス	表示される情報
CIRCUITS	OpenVMS Cluster システムの仮想サーキットに関する情報。
CLUSTER	OpenVMS Cluster システムに関する一般的な情報。たとえば、そのクラスタが形成された時刻、システムがクラスタに追加された、またはクラスタから削除された最新の日付、クラスタ・ウォラムなど。
CONNECTIONS	OpenVMS Cluster システム内の仮想サーキットで確立した接続に関する情報。
COUNTERS	各接続が確立されてからの累計トラフィック。
CREDITS	OpenVMS Cluster システム内の各接続における送信クレジット・カウントおよび受信クレジット・カウント。
ERRORS	各ポートで発生したエラーの数とポートの再初期化の実行可能性に関する情報。
LOCAL_PORTS	OpenVMS Cluster システムに対するローカル・システム・インタフェースの情報。たとえば、各ポートの名前、番号、状態、および各ポートに関連するキューに登録されたエントリの数。

表 8-3 SHOW CLUSTER レポートで入手可能なクラス情報 (続き)

クラス	表示される情報
MEMBERS	OpenVMS Cluster システムにアクティブに参加しているシステムの情報。
SYSTEMS	全 OpenVMS Cluster システムの情報。ノード名、識別番号、ハードウェア・タイプ、ソフトウェア・バージョンが示される。

例

SHOW CLUSTER レポートに CLUSTER クラスを追加する場合は、次のコマンドを入力します。

Command> **ADD CLUSTER**

結果として、画面に表示される内容は 図 8-2 「CLUSTER が表示されている SHOW CLUSTER レポート」に示すように変化します。CLUSTER クラスは、省略時の SHOW CLUSTER レポートの下に表示されます。

図 8-2 CLUSTER が表示されている SHOW CLUSTER レポート

View of Cluster from system ID 65536 node: CLUB 31DEC1997 14:00:00

SYSTEMS		MEMBERS
NODE	SOFTWARE	STATUS
CLUB HSJ400 HSC900 CHIP DISK3 DISK1 SPREE SPRITZ	VMS V7.2 HSJ V25J HSC V860 VMS V7.1 RFX V256 RFX V256 VMS V6.2 VMS V7.1	MEMBER MEMBER MEMBER MEMBER

CLUSTER						
CL_EXP	CL_QUORUM	CL_VOTES	QF_VOTE	CL_MEMBERS	FORMED	LAST_TRANSITION
3	2	3	NO	4	15JUN1997	10DEC1997

ZK8999AGE

CLUSTER クラスのフィールドの説明については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の SHOW CLUSTER の節を参照してください。

8.3.4 表示データの制御

SHOW CLUSTER コマンドにより、レポートからのフィールドまたはクラスの削除、画面からのブロードキャスト・メッセージの削除、任意のタイミングでの画面表示のリフレッシュができます。次の項では、これらの操作手順を説明します。

8.3.4.1 データ表示中のコマンド入力

SHOW CLUSTER の連続表示セッション中は、さまざまなコマンドを入力して、表示されているデータを変更することができます。『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』では、SHOW CLUSTER のコマンドについて詳しく説明しています。キーボードから入力を始めると、すぐに表示内容の更新が停止します。コマンドを入力して Return キーを押すと、表示内容の更新が再開され、別のコマンドを入力するまで続けられません。

省略時の設定では、更新は 15 秒間隔で行われます。その 15 秒間に新しいコマンドを入力しないとコマンド・プロンプトは消え、その場所に 2 行分のデータが表示されます。

8.3.4.2 ブロードキャスト・メッセージの削除

SHOW CLUSTER の連続セッション中にシステムからのブロードキャスト・メッセージを受信すると、受信したメッセージが画面の最下部に表示されます。メッセージが複数行にまたがる場合には、必要な数の行が使用されます。

作業方法

最新のブロードキャスト・メッセージは、確認が行われるまで画面上に残ります。確認は次のいずれかの方法で行います。

- Return キーを押す。
- Ctrl/W を押して画面をリフレッシュする。
- 何らかのコマンドを入力する。

複数のブロードキャスト・メッセージを受信している場合、2 番目のメッセージは次の更新が行われるまで画面に表示されません。

また、画面の最下部には、SHOW CLUSTER からのエラー・メッセージも表示されます。SHOW CLUSTER からのエラー・メッセージについては、『OpenVMS System Messages: Companion Guide for Help Message Users』を参照してください。

8.3.4.3 画面のリフレッシュ

通常、連続表示は、省略時の間隔または指定された間隔で更新またはリフレッシュされます。SHOW CLUSTER は、ソフトウェア・データベースを検索し、各フィールドのデータを抽出して保存し、新しいデータまたは変更されたデータがあれば表示し、時刻を更新します。HP ターミナルまたは HP 互換ターミナルでは、変更されたデータが反転表示されます。

作業方法

次のいずれかの方法を使用すると、いつでも画面をリフレッシュすることができます。

- ADD コマンド、REMOVE コマンド、INITIALIZE コマンド、または SET コマンドによって、表示の形式を変更する。
- REFRESH コマンドを使用する。
- Ctrl/W を押す。

8.3.5 表示データの書式化

SHOW CLUSTER ではフィールドおよびクラスを追加できるため、作成したレポートがターミナルの画面に表示しきれなくなることがあります。そのような場合には、必要に応じてレポートの書式を変更することができます。

変更方法	参照箇所
レポートからの情報の削除	8.3.5.1 項 「レポートからの情報の削除」
フィールドと画面サイズの変更	8.3.5.2 項 「フィールドと画面サイズの変更」
レポートの移動	8.3.5.3 項 「レポートの移動」
レポートのスクロール	8.3.5.4 項 「レポートのスクロール」

8.3.5.1 レポートからの情報の削除

レポートが画面に収まるように、フィールドまたはクラスを削除して、レポートの幅を小さくすることができます。また、省略時の設定で表示されるフィールドやクラスの中にはそれほど重要でないものもあります。特定の種類のデータを削除して、レポートを短くすることもできます。

作業方法

複数のフィールドまたはクラス全体を削除するには、REMOVE コマンドを使用します。1つのフィールドまたは1つのクラスを削除する場合には、REMOVE コマンドに適切な修飾子を指定して使用します。この形式で使用するクラス名および修飾子については『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』の SHOW CLUSTER の節を参照してください。

例

1.

```
Command> REMOVE SOFTWARE
```

このコマンドは、図 8-1 「省略時の SHOW CLUSTER レポート」で示した SHOW CLUSTER レポートから SOFTWARE フィールドを削除する。

使用可能なフィールド名については、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』の SHOW CLUSTER の説明を参照。

2.

```
Command> REMOVE MEMBERS
```

このコマンドは、図 8-1 「省略時の SHOW CLUSTER レポート」で示した SHOW CLUSTER レポートから MEMBERS クラスを削除する。

8.3.5.2 フィールドと画面サイズの変更

レポートが画面に収まるようにするため、レポートの特定のフィールドの幅を小さくすることができます。たとえば、考えられるすべての値が入るようにその幅が設定されているフィールドでも、実際にはそれほどのスペースが必要ない場合があります。そのようなときは、SET (フィールド) コマンドを使用して、そのフィールドの幅を調整することができます。

また、SHOW CLUSTER では、ターミナルの画面のサイズを調整することができます。ターミナルが HP 互換で幅の広いレポートをサポートしている場合、SET SCREEN コマンドに適切な値を指定することにより、画面の幅を最大 511 カラムまで増やすことができます。

例

1.

```
Command> SET TRANSITION_TYPE/WIDTH=10
```

この例は、TRANSITION_TYPE フィールドの幅を 10 に設定する。その結果、フィールドには日付だけが表示され、時刻は表示されなくなる。

2.

```
Command> SET SCREEN=132
```

画面の幅を 132 に設定する。

SET (フィールド) コマンドおよび SET SCREEN コマンドについての詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』を参照してください。

8.3.5.3 レポートの移動

省略時の設定では、SHOW CLUSTER ユーティリティは AUTO_POSITIONING ON で動作します。すなわち、画面のスペースをできるだけ有効に利用するように、SHOW CLUSTER によってレポートの位置が自動的に調整されます。しかし、MOVE コマンドを使用すると、レポートの位置を手動で調整することができます。MOVE コマンドを使用すると、AUTO_POSITIONING は暗黙に OFF に設定されます。

画面に複数のレポートを表示させる場合は、まず位置を変更するレポートを選択します。SELECT **ウィンドウ名** コマンドに位置を変更するレポートの名前を指定します。たとえば、次のようにします。

- SCS (省略時のレポート。通常、SYSTEMS クラスおよび MEMBERS クラスのフィールドが含まれる。)

- CLUSTER
- LOCAL_PORTS



注意:

省略時の SCS レポートに含まれないクラスを選択するとき、そのクラスが画面に含まれていない場合は、SELECT コマンドを実行する前にそのクラスを画面に表示させる必要があります。たとえば、次のコマンド行を入力します。

```
Command> ADD LOCAL_PORTS
```

別の方法として、Select ファンクション・キーあるいはキーパッド上のピリオド・キーを繰り返し押し出すことによって、レポートを順番に表示させることもできます。選択されたレポートは強調表示されます。

作業方法

レポートを移動する場合は、次のいずれかの操作を行います。

- Command> プロンプトに対して MOVE コマンドを入力する。
- MOVE コマンドとして定義した矢印キーを使用する。

```
Command> SET FUNCTION MOVE
```

このコマンドを実行すると、矢印キーは次のように再定義される。

矢印キー	対応するコマンド
上向き矢印	MOVE UP 1
下向き矢印	MOVE DOWN 1
右向き矢印	MOVE RIGHT 1
左向き矢印	MOVE LEFT 1

MOVE コマンドを入力すると、表示部分はカラム単位 (水平方向)、あるいは行単位 (垂直方向) にその位置を変える。たとえば、コマンド MOVE LEFT 5 を入力すると、表示される部分は 5 カラム分左に移動する。この時、新たに表示された部分にデータは表示されない。

レポートの位置が決まったら DESELECT コマンドを入力する。このコマンドを入力すると、レポートは新しい位置に移動する。また、このコマンドを入力する前に別の SELECT コマンドを入力すると、それまでの MOVE 操作は確定され、レポートは新しい位置に移動する。

例

```
Command> SELECT CLUSTER  
Command> MOVE RIGHT 10  
Command> DESELECT
```

この例の各コマンドの意味は次のとおりです。

1. SELECT コマンドにより CLUSTER レポートを選択する (強調表示される)。
2. MOVE コマンドにより、レポートのフレームを 10 カラム分右に移動する。
3. DESELECT コマンドにより、MOVE 操作を終了し、レポートの内容を表示する。

詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』の SELECT, SET FUNCTION, および DESELECT の各コマンドの説明を参照してください。

作業が終了したら、次のコマンドを入力して矢印キーの設定を元に戻しておきます。

Command> SET FUNCTION EDIT

8.3.5.4 レポートのスクロール

SCROLL コマンドを使用すると、カラム・ヘッダを表示したままレポートをスクロールさせることができます。このコマンドにより、レポートをフィールド単位で水平方向に移動したり、行単位で垂直方向に移動することができます。垂直方向にスクロールする場合、カラム・ヘッダは固定されたままになります。

画面に複数のレポートが表示されている場合は、最初に SELECT コマンドを入力してスクロール対象のレポートを選択する必要があります。選択されたレポートは強調表示されます。

作業方法

レポートをスクロールするためには、次のいずれかの方法を使用します。

- コマンド・プロンプトで SCROLL コマンドを入力する。
- SCROLL コマンドとして定義した矢印キーを使用する。

Command> SET FUNCTION SCROLL

このコマンドにより、矢印キーは次のように再定義される。

矢印キー	対応するコマンド
上向き矢印	SCROLL UP 1
下向き矢印	SCROLL DOWN 1
右向き矢印	SCROLL RIGHT 1
左向き矢印	SCROLL LEFT 1

例

Command> SELECT SCS

Command> SET FUNCTION SCROLL

この例では、まず SCS レポートを選択して強調表示し、次に矢印キーをスクロール機能に設定します。詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』の SET FUNCTION コマンドおよび SCROLL コマンドの説明を参照してください。

作業が終了したら、次のコマンドを入力して矢印キーの設定を元に戻しておきます。

Command> SET FUNCTION EDIT

8.3.6 SHOW CLUSTER スタートアップ初期化ファイルの作成

スタートアップ初期化ファイルを作成して、SHOW CLUSTER の表示内容をカスタマイズすることができます。SHOW CLUSTER を起動すると、最初にこのファイルが実行されます。SHOW CLUSTER ユーティリティは元の省略時の表示内容から、このファイルの指定に従ってクラスおよびフィールドを追加または削除します。この結果の表示形式が、ユーザの表示形式になります。スタートアップ初期化ファイルの書式は次のとおりです。

```
!  
!Startup Initialization File  
!  
!  
INITIALIZE  
REMOVE MEMBERS  
ADD RP_REVISION,RP_TYPE,SYS_ID  
SET SCREEN=132
```

このスタートアップ・プロシージャ (スタートアップ初期化ファイル) を使用すると、SHOW CLUSTER は省略時の表示内容から MEMBERS の情報を削除します。さらに、CIRCUITS クラスの RP_REVISION フィールドと RP_TYPE フィールド、および SYSTEMS クラスの SYS_ID フィールドを追加します。このプロシージャの最後の行により、画面の幅が 132 カラムに設定されます。

作業方法

初期化ファイルの作成手順は次のとおりです。

1. 論理名 SHOW_CLUSTER\$INIT をデバイス:[ディレクトリ]SHCINI と定義した後、SHOW CLUSTER を起動する。

表示が開始される前にスタートアップ・ファイルを実行する場合は、論理名 SHOW_CLUSTER\$INIT を初期化ファイルに設定する。たとえば、次のコマンド行を入力する。

```
DEFINE SHOW_CLUSTER$INIT DEVA:[JONES]SHCINI
```

SHOW CLUSTER はその起動時に SHOW_CLUSTER\$INIT で定義されたファイルを検索する。この例では、SHOW CLUSTER は起動時に DEVA:[JONES]SHCINI.INI を探す。初期化ファイルが見つかったら、SHOW CLUSTER は表示を開始する前にプロシージャを実行する。

SHOW_CLUSTER\$INIT が定義されていない場合、または定義にディレクトリ指定が含まれていない場合は、SHOW CLUSTER は現在の省略時のディレクトリで SHOW_CLUSTER.INI と名付けられたファイルを探す。

2. 連続表示セッション中に SHOW CLUSTER コマンドを使用して、表示内容をカスタマイズする。
3. 次のコマンドを入力してコマンド・シーケンスを保存する。

```
Command> SAVE SHOW_CLUSTER$INIT.INI
```

SHOW_CLUSTER\$INIT.INI は必ず指定する。このファイル名を省略すると、SAVE コマンドは省略時の設定で .COM ファイルを作成するので、SHOW CLUSTER ユーティリティは初期化ファイルを見つけることができない。SHOW CLUSTER ユーティリティは、初期化ファイルの検索時にタイプが .INI のファイルを探す。

SAVE コマンドによって作成されたファイルにコメントを加えて読みやすくすることができます。詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル(下巻)』の SAVE コマンドの説明を参照してください。

SHOW CLUSTER ユーティリティを使用しないで、コマンド・プロシージャを作成するのと同じ方法で、初期化ファイルを作成することができます。その手順を次の項で説明します。

8.3.7 SHOW CLUSTER ユーティリティのコマンドを含むコマンド・プロシージャの使用

SHOW CLUSTER のコマンドを含むコマンド・プロシージャを作成することができます。そのようなファイルを使用すれば、会話形式でコマンドを入力しないで表示特性を変更することができます。SHOW CLUSTER ユーティリティの連続表示セッション中にこのコマンド・プロシージャを使用して、たとえば出力をカスタマイズするための一連のコマンドを実行することができます。

次に、SHOW CLUSTER のコマンドを含むコマンド・プロシージャを作成するためのガイドラインを示します。

- SHOW CLUSTER の正しいコマンドを使用する。
- コマンド・プロシージャのネストは 16 レベルまでとする。

- ファイルの最初のコマンドは SHOW CLUSTER の INITIALIZE にする。INITIALIZE コマンドが実行されると、レポートの内容を変更するためのコマンドが実行される前に、そのレポートが確実に「既知」の状態になる。



注意:

コマンド・プロシージャに EXIT コマンドを記述しないでください。プロシージャの最後にも記述することはできません。EXIT コマンドがあると SHOW CLUSTER ユーティリティは終了し、レポートの内容を見る前にその内容が消去されます。

また、SHOW CLUSTER のコマンド・プロシージャをバッチ・ジョブから実行しないでください。

次に、レポートの表示項目をカスタマイズするコマンド・プロシージャの例を示します。

```
!  
! Include only the node field from the default display; show votes  
! and quorum for each node and for the cluster as a whole.  
!  
INITIALIZE  
REMOVE SOFTWARE, STATUS  
ADD VOTES, QUORUM, CL_VOTES, CL_QUORUM
```

このコマンド・プロシージャは、レポートから SOFTWARE フィールドと STATUS フィールドとを削除し、クラスタのクォーラムおよびポートに関する情報を示すフィールドを追加します。

SHOW CLUSTER の連続表示セッション中にコマンド・プロシージャを実行する場合は、プロシージャ実行コマンド (@) とともにプロシージャのファイル名を指定します。コマンド・プロシージャの省略時のファイル・タイプは .COM です。

例

SYSMOD.COM という名前のコマンド・プロシージャを実行します。

```
Command> @SYSMOD
```

この例ではファイル・タイプが省略されているため、ファイル・タイプ .COM が使用されます。

コマンド・プロシージャの作成についての詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』の SAVE コマンドの説明を参照してください。

8.4 SYSMAN ユーティリティと OpenVMS Cluster 管理について

SYSMAN では、次の 2 種類の方法で OpenVMS Cluster 管理をサポートしています。

- クラスタ固有コマンドの CONFIGURATION SET および CONFIGURATION SHOW。クラスタ内の機密保護データとシステム時刻を管理するときに使用する。
- DO コマンドによる DCL レベルのコマンドへのアクセス。この機能により、クラスタ全体に単一の DCL コマンドを適用することができ、各ノード上でコマンドを入力する必要はなくなる。

SYSMAN のコマンドを使用するためには、適切な特権が必要になります。各コマンドについての詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』を参照してください。

8.5 システム管理ユーティリティ (SYSMAN) による機密保護の管理

システム管理ユーティリティ (SYSMAN) の CONFIGURATION コマンドにより、OpenVMS Cluster システムの機密保護データを管理することができます。表 8-4 「CONFIGURATION コマンド」に CONFIGURATION 関連のコマンドとそれらの機能をまとめます。

表 8-4 CONFIGURATION コマンド

コマンド	機能
CONFIGURATION SET CLUSTER_AUTHORIZATION	ローカル・エリア・クラスタのグループ番号とパスワードを変更する。
CONFIGURATION SHOW CLUSTER_AUTHORIZATION	ローカル・エリア・クラスタのグループ番号とマルチキャスト・アドレスを表示する。

8.5.1 グループ番号およびパスワードの変更

グループ番号は、クラスタ内のノードのグループを識別し、対応するイーサネット・アドレスは、メッセージをクラスタ内の全ノードに送信するために使用されます。OpenVMS Cluster パスワードにより、クラスタ・メンバシップの統一性が保護されます。

CONFIGURATION SET CLUSTER_AUTHORIZATION コマンドを使うと、SYS\$SYSTEM:CLUSTER_AUTHORIZE.DAT に記録されているグループ番号とパスワードが変更されます。通常、CLUSTER_AUTHORIZE.DAT ファイル内のレコードを変更する必要はありません。

複数のシステム・ディスクがシステム構成に含まれている場合、SET ENVIRONMENT/CLUSTER コマンドによって環境がクラスタとして定義されていれば、SYSMAN は CLUSTER_AUTHORIZE.DAT の各コピーを自動的に更新します。



重要:

グループ番号およびパスワードのいずれかを変更した場合は、クラスタ全体をリブートする必要があります。

機密保護上の理由から、クラスタのパスワードを表示することはできませんが、CONFIGURATION SHOW CLUSTER_AUTHORIZATION コマンドを使用すると、グループ番号およびグループ・マルチキャスト・アドレスを表示することができます。

例

1. 次の例では、環境を特定のクラスタに設定し、SYSPRV 特権を獲得し、クラスタ・パスワードを変更している。

```

SYSMAN> SET ENVIRONMENT/CLUSTER/NODE=NODE21
SYSMAN> SET PROFILE/PRIVILEGE=SYSPRV
SYSMAN> CONFIGURATION SET CLUSTER AUTHORIZATION/PASSWORD=GILLIAN
%SYSMAN-I-CAFOLDGROUP, existing group will not be changed
%SYSMAN-I-GRPNCHG, Group number not changed
SYSMAN-I-CAFREBOOT, cluster authorization file updated.
The entire cluster should be rebooted.

```

2. 次の例では、ノード NODE21 のグループ番号とマルチキャスト・アドレスを表示している。クラスタ内の他のノード上でも同じグループ番号およびパスワードが使用されるため、それ以上の情報は表示されない。

```

SYSMAN> CONFIGURATION SHOW CLUSTER AUTHORIZATION
Node NODE21: Cluster group number 65240
Multicast address: AB-00-04-01-F2-FF

```

8.6 SYSMAN DO コマンドによる OpenVMS Cluster の管理

SYSMAN の DO コマンドを使用すると、現在の環境のすべてのノード上で、DCL コマンドおよびコマンド・プロシージャを実行することができます。このコマンドは、OpenVMS Cluster

内のノード上でシステム管理作業をルーチン化して行う場合に便利です。たとえば、次の作業が挙げられます。

- イメージのインストール
- ソフトウェアのスタートアップ
- デバイスのチェック
- メモリのチェック

各 DO コマンドは独立したプロセスとして実行されるため、DO コマンド相互間でプロセス・コンテキストが保持されることはありません。このため、DCL コマンドは必ず 1 つのコマンド文字列で表現します。また、入力データを必要とするプログラムを実行することはできません。

クラスタ環境において、SYSMAN はクラスタ内のすべてのノード上でコマンドを順次実行します。あるノード上で 1 つのコマンドの動作が完了するまで、SYSMAN は同じコマンドを次のノードに送信しません。コマンドを実行できないノードがあると、そのノードからエラー・メッセージが返されます。コマンドを送信した先のノードから特定の時間内に応答がないと、SYSMAN はエラー・メッセージを表示します。

OpenVMS VAX と OpenVMS Alpha の両方を実行しているデュアル・アーキテクチャの異質 OpenVMS Cluster では、DO コマンドを使用する場合に特別な処置が必要な場合があります。たとえば、それぞれのアーキテクチャにおいて異なる名前を持つイメージをインストールする場合は、VAX ノードおよび Alpha ノードまたは 164 ノード用に論理名テーブルを作成すれば、DO コマンドを使用することができます。下記の例を参照してください。

DCL コマンドの中には MOUNT/CLUSTER や SET QUORUM/CLUSTER のように、クラスタ全体で動作するように設計されたものもあります。同様に、クラスタ単位の論理名や論理テーブルへの操作はクラスタ全体に操作されるように設計されています。環境がクラスタに設定されている場合には、このようなコマンドは SYSMAN の DO コマンドとともに使用しないようにしてください。SPAWN コマンドを使って SYSMAN を一時的に終了させ、DCL でこれらのコマンドを使用する、またはクラスタ内部で環境を単一ノードに定義するという方法もあります。

例

1. 次の例では、クラスタ上にイメージをインストールしている。まず、現在の特権に CMKRNL 特権および SYSPRV 特権を追加する。これらの特権は INSTALL コマンドおよび AUTHORIZE コマンドで必要になるものである。DO INSTALL コマンドは STATSHR ファイルをインストールする。DO MCR AUTHORIZE コマンドは、ユーザ Jones のアカウントを設定し、パスワード、省略時のデバイス、および省略時のディレクトリを指定する。

```
SYSMAN> SET PROFILE/PRIVILEGES=(CMKRNL,SYSPRV)/DEFAULT=SYS$SYSTEM
SYSMAN> DO INSTALL ADD/OPEN/SHARED WRKD$: [MAIN] STATSHR
SYSMAN> DO MCR AUTHORIZE ADD JONES/PASSWORD=COLUMBINE -
_SYSMAN> /DEVICE=WORK1/DIRECTORY=[JONES]
```

2. 次の例では、環境をクラスタに設定し、クラスタ内の各ノード上で XYZ というソフトウェア製品をスタートアップしている。

```
SYSMAN>SET ENVIRONMENT/CLUSTER
%SYSMAN-I-ENV, Current command environment:
      Clusterwide on local cluster
      Username SMITH      will be used on nonlocal nodes
SYSMAN> DO @SYS$STARTUP:XYZ_STARTUP
```

3. 次の例は、デュアル・アーキテクチャの異質クラスタ内の VAX ノード、Alpha ノード、または 164 ノードに論理名を定義して、DO コマンドを使ってアーキテクチャ固有のイメージをインストールできるようにする方法を示している。

```
$ CREATE/NAME_TABLE/PARENT=LNM$SYSTEM_DIRECTORY SYSMAN$NODE_TABLE
$ DEFINE/TABLE=SYSMAN$NODE_TABLE AXP_NODES NODE21,NODE22,NODE23
$ DEFINE/TABLE=SYSMAN$NODE_TABLE VAX_NODES NODE24,NODE25,NODE26
```

```

$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> SET ENVIRONMENT/NODE=AXP_NODES
%SYSMAN-I-ENV, current command environment:
    Individual nodes: NODE21,NODE22,NODE23
    Username BOUCHARD will be used on nonlocal nodes

SYSMAN> DO INSTALL REPLACE SYS$LIBRARY:DCLTABLES.EXE
%SYSMAN-I-OUTPUT, command execution on node NODE21
%SYSMAN-I-OUTPUT, command execution on node NODE22
%SYSMAN-I-OUTPUT, command execution on node NODE23
SYSMAN> DO INSTALL REPLACE SYS$SYSTEM:DEC_FORTRAN.EXE
%SYSMAN-I-OUTPUT, command execution on node NODE21
%SYSMAN-I-OUTPUT, command execution on node NODE22
%SYSMAN-I-OUTPUT, command execution on node NODE23

SYSMAN> SET ENVIRONMENT/NODE=VAX_NODES
%SYSMAN-I-ENV, current command environment:
    Individual nodes: NODE24,NODE25,NODE26
    Username BOUCHARD will be used on nonlocal nodes

SYSMAN> DO INSTALL REPLACE SYS$LIBRARY:DCLTABLES.EXE
%SYSMAN-I-OUTPUT, command execution on node NODE24
%SYSMAN-I-OUTPUT, command execution on node NODE25
%SYSMAN-I-OUTPUT, command execution on node NODE26
SYSMAN> DO INSTALL REPLACE SYS$SYSTEM:FORTRAN$MAIN.EXE
%SYSMAN-I-OUTPUT, command execution on node NODE24
%SYSMAN-I-OUTPUT, command execution on node NODE25
%SYSMAN-I-OUTPUT, command execution on node NODE26

```

4. DISK2 でオープンされているファイルを表示している。DISK2 をディスマウントしようとする場合、このコマンドによってクラスタ内のどのユーザがファイルをオープンしているかを調べる。

```

SYSMAN >SET ENVIRONMENT/CLUSTER
%SYSMAN-I-ENV, Current command environment:
    Clusterwide on local cluster
    Username SMITH will be used on nonlocal nodes
SYSMAN> DO SHOW DEVICE/FILES DISK2:

%SYSMAN-I-OUTPUT, command execution on node NODE21
Files accessed on device $1$DIA2: (DISK2, NODE22) on 14-may-2000 15:44:06.05
Process name      PID      File name
                  00000000 [000000]INDEXF.SYS;1
%SYSMAN-I-OUTPUT, command execution on node NODE22
Files accessed on device $1$DIA2: (DISK2, NODE21) on 14-may-2000 15:44:26.93
Process name      PID      File name
                  00000000 [000000]INDEXF.SYS;1
%SYSMAN-I-OUTPUT, command execution on node NODE23
Files accessed on device $1$DIA2: (NODE21, NODE22) on 14-may-2000 15:45:01.43
Process name      PID      File name
                  00000000 [000000]INDEXF.SYS;1
%SYSMAN-I-OUTPUT, command execution on node NODE24
Files accessed on device $1$DIA2: (NODE22, NODE21) on 14-may-2000 15:44:31.30
Process name      PID      File name
                  00000000 [000000]INDEXF.SYS;1
Susan Scott      21400059 [SCOTT]DECW$SM.LOG;228
_FTA7:          214000DD [SCOTT]CARE_SDML.TPU$JOURNAL;1
%SYSMAN-I-OUTPUT, command execution on node NODE25
Files accessed on device $1$DIA2: (NODE21, NODE22) on 14-may-2000 15:44:35.50
Process name      PID      File name
                  00000000 [000000]INDEXF.SYS;1
DECW$SESSION    226000E6 [SNOW]DECW$SM.LOG;6
_FTA17:         2260009C [SNOW.MAIL]MAIL.MAI;1
SNOW_1          2260012F [SNOW.MAIL]MAIL.MAI;1
SNOW_2          22600142 [SNOW.MAIL]MAIL.MAI;1
SNOW_3          22600143 [SNOW.MAIL]MAIL.MAI;1

```

5. 次の例では、クラスタを構成するノードで利用可能なメモリ容量を表示している。ソフトウェアをインストールする場合は、このコマンドにより各ノードで利用できるメモリの容量が十分かどうかを調べる。

```

SYSMAN > SET ENVIRONMENT/NODE=(NODE21,NODE22)
%SYSMAN-I-ENV, Current command environment:
      Clusterwide on local cluster
      Username SMITH will be used on nonlocal nodes

SYSMAN> DO SHOW MEMORY
%SYSMAN-I-OUTPUT, command execution on node NODE21
      System Memory Resources on 14-may-2000 15:59:21.61
Physical Memory Usage (pages):   Total      Free      In Use   Modified
Main Memory (64.00Mb)           131072    63955    65201    1916
Slot Usage (slots):              Total      Free      Resident Swapped
Process Entry Slots              360       296      64        0
Balance Set Slots                324       262      62        0
Fixed-Size Pool Areas (packets): Total      Free      In Use   Size
Small Packet (SRP) List          10568     1703     8865     128
I/O Request Packet (IRP) List    3752     925      2827     176
Large Packet (LRP) List          157       28       129     1856
Dynamic Memory Usage (bytes):    Total      Free      In Use   Largest
Nonpaged Dynamic Memory          1300480   97120    1203360  60112
Paged Dynamic Memory             1524736  510496   1014240  505408
Paging File Usage (pages):       Free      Reservable Total
DISK$MTWAIN_SYS: [SYS0.SYSEXEXE] SWAPFILE.SYS
                                   10000     10000    10000
DISK$MTWAIN_SYS: [SYS0.SYSEXEXE] PAGEFILE.SYS
                                   60502     -52278   100000
Of the physical pages in use, 19018 pages are permanently allocated to VMS.

%SYSMAN-I-OUTPUT, command execution on node NODE22
      System Memory Resources on 14-may-2000 15:59:42.65
Physical Memory Usage (pages):   Total      Free      In Use   Modified
Main Memory (32.00Mb)           65536     44409    20461    666
Slot Usage (slots):              Total      Free      Resident Swapped
Process Entry Slots              240       216      24        0
Balance Set Slots                212       190      22        0
Fixed-Size Pool Areas (packets): Total      Free      In Use   Size
Small Packet (SRP) List          5080     2610     2470     128
I/O Request Packet (IRP) List    3101     1263     1838     176
Large Packet (LRP) List          87        60       27     1856
Dynamic Memory Usage (bytes):    Total      Free      In Use   Largest
Nonpaged Dynamic Memory          1165312  156256   1009056  114432
Paged Dynamic Memory             1068032  357424   710608   352368
Paging File Usage (pages):       Free      Reservable Total
DISK$MTWAIN_SYS: [SYS1.SYSEXEXE] SWAPFILE.SYS
                                   10000     10000    10000
DISK$MTWAIN_SYS: [SYS1.SYSEXEXE] PAGEFILE.SYS
                                   110591     68443   120000
Of the physical pages in use, 9056 pages are permanently allocated to VMS.

```


第9章 ネットワークについて

この章では、次の内容について説明します。

- OpenVMS システムのための TCP/IP および DECnet ネットワーク・ソフトウェア・オプション
- 使用している OpenVMS システムに適したネットワーク・ソフトウェア・オプションの決定方法
- 使用しているシステムをネットワークに参加させる準備
- 選択したネットワーク・ソフトウェアをインストールし、構成し、管理するために役立つ詳細情報の入手先

この章では、それぞれの簡単な紹介にとどめています。それぞれの計画、インストール、構成、使用、および管理については、使用するネットワーク製品のマニュアルを参照してください。

この章で説明する内容

この章では、次の作業について説明します。

作業	参照箇所
使用しているシステムに適したネットワーク・ソフトウェアの選択	9.2 項 「HP ネットワーク・ソフトウェアの選択」
TCP/IP ネットワークに参加するための準備	9.4 項 「TCP/IP ネットワークに参加するための準備」
TCP/IP Services のインストールおよび構成	9.5 項 「TCP/IP Services のインストールおよび構成」
TCP/IP Services の開始および終了	9.6 項 「TCP/IP Services の開始と停止」
DECnet-Plus ネットワークに参加するための準備	9.9 項 「DECnet-Plus ネットワークに参加するための準備」
DECnet-Plus のインストールおよび構成	9.10 項 「DECnet-Plus のインストールおよび構成」
DECnet Phase IV から DECnet-Plus への移行	9.12 項 「DECnet Phase IV から DECnet-Plus への移行」
DECnet-Plus の開始および終了	9.13 項 「DECnet-Plus の開始と停止」
IP ネットワーク・バックボーンを介した DECnet アプリケーションの実行	9.11 項 「DECnet over TCP/IP の使用方法」

この章では、次の概念について説明します。

概念	参照箇所
OpenVMS システムのためのネットワーク・ソフトウェア・オプション	9.1 項 「OpenVMS ネットワーク・ソフトウェア・オプション」
HP TCP/IP Services for OpenVMS ソフトウェアの概要	9.3 項 「HP TCP/IP Services for OpenVMS について」
DECnet-Plus for OpenVMS ソフトウェアの概要	9.8 項 「DECnet-Plus for OpenVMS ネットワーク・ソフトウェア」
DECnet over TCP/IP の使用方法	9.11 項 「DECnet over TCP/IP の使用方法」

9.1 OpenVMS ネットワーク・ソフトウェア・オプション

OpenVMS システムを他のシステムおよびネットワークに接続できる能力は、OpenVMS オペレーティング・システムの機能の基本的な部分です。

OpenVMS システムで使用可能なネットワーク・ソフトウェア・オプションは、次のとおりです。

- HP TCP/IP Services for OpenVMS
- 他社製の OpenVMS 用 TCP/IP ソフトウェア



注意:

他社製のネットワーク・ソフトウェアの構成方法については、その会社の製品ドキュメントを参照してください。

- HP DECnet-Plus for OpenVMS (Phase V)
- HP DECnet for OpenVMS (Phase IV)

これらのネットワーク・オプションのうち 1 つを OpenVMS にインストールして使用することも、複数のオプションを組み合わせてインストールして使用し、次のような作業を実行することが可能です。

- 他の OpenVMS システムとリソースを相互操作し、共用する。
- UNIX® や Windows NT® など、他のオペレーティング・システムを実行しているシステムとリソースを相互操作し、共用する。
- DECnet プロトコルおよびアプリケーションを、TCP/IP プロトコルおよびアプリケーションと組み合わせて使用する。

9.2 HP ネットワーク・ソフトウェアの選択

表 9-1 「OpenVMS システム用の HP ネットワーク・ソフトウェアの選択」で、OpenVMS のための HP レイヤード・ネットワーク・ソフトウェアについて説明します。これらのソフトウェアは、OpenVMS オペレーティング・システムのインストールまたはアップグレード中に、インストールすることができます。また、個別にレイヤード・プロダクトとしてインストールすることも可能です。

表 9-1 OpenVMS システム用の HP ネットワーク・ソフトウェアの選択

製品	説明
HP TCP/IP Services for OpenVMS	<p>TCP/IP Services は、弊社が業界標準の Transmission Control Protocol/Internet Protocol (TCP/IP) プロトコル・セットを OpenVMS VAX システム、Alpha システム、および I64 システム用にインプリメントしたものです。OpenVMS システムが他の異機種混合ネットワーク (インターネット、UNIX システム、Windows NT システムなど) と通信する必要がある場合には、HP TCP/IP Services for OpenVMS ソフトウェアを選択します。TCP/IP Services を使用すると、遠隔ホストに接続して、ファイルにアクセスしたり、メッセージを交換したり、アプリケーションを開発したり、ネットワークを監視したり、その他の重要な作業を実行したりできます。</p> <p>TCP/IP Services は、システムで唯一のネットワーク・ソフトウェアとして使用できます。OpenVMS では、DECnet ソフトウェアを使用することが必須ではありません。</p> <p>TCP/IP プロトコルは、DECnet-Plus プロトコルと共存することができます。TCP/IP Services をネットワークのバックボーンとしてインストールしながら、DECnet-Plus もインストールし、DECnet のアプリケーションおよび機能を引き続き使用することができます。DECnet プロトコルを IP バックボーンを介して使用する方法については、9.11 項「DECnet over TCP/IP の使用方法」を参照してください。</p> <p>TCP/IP Services ソフトウェアのインストール、使用、管理の概要については、9.3 項「HP TCP/IP Services for OpenVMS について」を参照してください。</p>
DECnet-Plus for OpenVMS (Phase V)	<p>DECnet-Plus は、弊社が Digital Network Architecture (DNA) の Phase V をインプリメントしたものです。このソフトウェアでは、以前の DECnet Phase IV 製品との完全な下位互換性の他、DECnet (NSP) および OSI を、DECnet、OSI、または TCP/IP ネットワーク・バックボーンを介して実行する機能も提供されています。IP バックボーンを介して DECnet および OSI プロトコルを使用する方法については、9.11 項「DECnet over TCP/IP の使用方法」を参照してください。</p> <p>OpenVMS システムが DECnet (Phase IV および Phase V) または OSI アプリケーションおよびプロトコルを使用して通信する必要があり、システムで TCP/IP または OSI プロトコルあるいはその両方と共存する必要がある場合には、DECnet-Plus ソフトウェアを選択します。</p> <p>DECnet Phase IV から徐々にネットワークをアップグレードしている場合には、Phase IV 機能を OSI の機能および利点とともに使用できるため、DECnet-Plus の機能が役立ちます。</p> <p>DECnet-Plus ソフトウェアのインストール、使用、管理の概要については、9.8 項「DECnet-Plus for OpenVMS ネットワーク・ソフトウェア」を参照してください。</p> <p>注意: DECnet-Plus には、DNA Phase IV および Phase V プロトコルの両方が含まれているため、DECnet-Plus をインストールする同じシステム上で別の DECnet Phase IV ソフトウェア製品を実行することはできません。</p> <p>DECnet-Plus へのアップグレードについては、『DECnet-Plus Planning Guide』を参照してください。</p>
DECnet for OpenVMS (Phase IV)	<p>DECnet Phase IV は、弊社が Digital Network Architecture (DNA) の Phase IV をインプリメントしたものです。この製品には、DECnet-Plus (Phase V) で後に提供される OSI プロトコルや TCP/IP 通信機能が含まれていません。TCP/IP Services ソフトウェアは、DECnet Phase IV とともにシステムで実行できますが、IP をネットワーク・バックボーンとして実行することはできません。</p> <p>異機種混合環境ではなく、従来型の OpenVMS 環境で通信するために OpenVMS を使用する場合には、従来の DECnet 製品である DECnet Phase IV を選択する方がよい場合があります。</p> <p>この章には、従来の DECnet Phase IV 製品について製品情報の概要は含まれていません。DECnet Phase IV の使用方法については、『DECnet for OpenVMS Networking Manual』および DECNET-VMS-MGMT-UTIL-REF;の各ドキュメントを参照してください (OpenVMS Documentation CD-ROM に含まれています)。</p>

表 9-2 「HP ネットワーク・ソフトウェアの相互操作性オプション」に、複数の OpenVMS システム間の通信のための、可能なネットワーク・ソフトウェアの組み合わせを示します。

表 9-2 HP ネットワーク・ソフトウェアの相互操作性オプション

システム A のソフトウェア	システム B のソフトウェア	システム A と B の間の通信に使用できるソフトウェア
TCP/IP Services	TCP/IP Services	TCP/IP アプリケーション
DECnet Phase IV	DECnet Phase IV	DECnet アプリケーション
DECnet-Plus	DECnet-Plus	DECnet アプリケーション、OSI アプリケーション
DECnet-Plus	DECnet Phase IV	DECnet アプリケーション
DECnet-Plus	OSI	OSI アプリケーション
TCP/IP Services および DECnet Phase IV	TCP/IP Services	TCP/IP アプリケーション
TCP/IP Services および DECnet Phase IV	DECnet Phase IV	DECnet アプリケーション
TCP/IP Services および DECnet-Plus	TCP/IP Services	TCP/IP アプリケーション
TCP/IP Services および DECnet-Plus	DECnet-Plus	DECnet アプリケーション、OSI アプリケーション
TCP/IP Services および DECnet-Plus	TCP/IP Services および DECnet-Plus	OSI アプリケーション、DECnet アプリケーション、DECnet アプリケーション、DECnet を TCP/IP を介して使用、OSI アプリケーション、OSI を TCP/IP を介して使用、TCP/IP アプリケーション
TCP/IP Services および DECnet-Plus	OSI (RFC 1006 をサポートする) および TCP/IP Services	OSI アプリケーション、OSI を TCP/IP で使用、TCP/IP アプリケーション
TCP/IP Services および DECnet-Plus	OSI (RFC 1006 をサポートしない) および TCP/IP Services	OSI アプリケーション、TCP/IP アプリケーション

9.3 HP TCP/IP Services for OpenVMS について

HP TCP/IP Services for OpenVMS TCP/IP Services 製品は、業界標準の TCP/IP 通信プロトコル・セットを、Internet Engineering Task Force (IETF) によって使用される Request for Comments (RFCs) の指定に従って、OpenVMS にインプリメントしたものです。

TCP/IP Services を使用すると、異機種混合ネットワークがインターコネクトできるようになるため、ユーザは次のようにさまざまな方法で遠隔ホストに接続することが可能になります。

- ネットワーク・ファイル・アクセス。遠隔ホスト上のファイルにアクセスできる。
- 電子メール。ホスト間でメッセージを交換できる。
- アプリケーション開発。アプリケーション・プログラマが、ローカル・ホストと遠隔ホストの間の通信のための TCP/IP クライアント/サーバ・アプリケーションを開発できる。
- ダウンロードおよびファイル転送。ホスト間でファイルを交換できる。
- ユーザ情報。同じローカル・ホストまたは遠隔ホストにログオンしている他のユーザに関する情報にアクセスできる。
- リモート管理。システム管理者が、遠隔ホストからネットワークおよびアプリケーションを監視できる。
- 遠隔ターミナル・アクセス。あたかも自分のターミナルが直接ホストに接続されているかのように、ホストにアクセスできる。
- 遠隔コマンドの実行。遠隔ホストに対してコマンドを実行できる。
- 遠隔プリント。プリント・ジョブを遠隔プリンタに対して送信したり、遠隔プリンタから受信できる。
- 遠隔ファイル・コピー。遠隔ホストにあるファイルをコピーできる。

- 遠隔ブート。サーバは遠隔クライアントにブート情報を提供できる。

TCP/IPによる相互接続では、それぞれのネットワークのハードウェアの詳細を意識せず、コンピュータは個々の物理的ネットワーク接続に関係なく通信することができます。TCP/IPは、標準の伝送手段と、全二重で信頼性が高いストリーム通信サービスの両方を、ソフトウェア・アプリケーションに提供します。

HP TCP/IP Services for OpenVMS ソフトウェアは、OpenVMS システム、UNIX システム、およびその他の TCP/IP プロトコル・セットと Sun Microsystems' Network File System (NFS) をサポートするシステム間の相互操作性およびリソース共有機能を提供します。TCP/IP システムおよびその他のインターネット・ホストは、イーサネット、Fiber Distributed Data Interface (FDDI)、トークン・リング、および非同期転送モード (ATM) などさまざまなネットワーク・ハードウェア構成で標準の TCP/IP プロトコルを使用して、データおよびリソースを共有しています。

TCP/IP ネットワークに接続されているそれぞれのエンド・システムは、**ホスト**と呼ばれます。それぞれのホストには、固有の名前とアドレスがあります。ローカル・ホストは現在使用しているシステムであり、遠隔ホストは通信している相手のシステムです。ホストは、ホスト間で情報を伝送する**ライン**によって接続されています。ラインは、データを1つのホストから別のホストへ渡すときに経由する物理パスです (ラインの例としては、電話ライン、光ファイバ・ケーブル、およびサテライトがあります)。

TCP/IP ネットワークは、パケット交換ネットワークと呼ばれます。情報は、連続したストリームではなく、小さなパケット単位でホストからホストへ伝送されます。たとえば、ホストから別のホストへ伝送されるファイルは、多数の小さなパケットに分割され、一度に1つずつ、ネットワークを介して送信されます。それぞれのパケットには、デスティネーション・ホストのアドレスに関する情報が含まれています。デスティネーションに着くと、パケットは組み立てられます。

データ・メッセージをソース・ホストからデスティネーション・ホストに出力先を指定するプロセスは、**ルーティング**と呼ばれます。相互に直接接続されていないホスト間では、仲介ホストを経由してデータをソースからデスティネーションに転送できます。

9.3.1 OpenVMS Cluster システムのサポート

HP TCP/IP Services for OpenVMS は、OpenVMS Cluster システム、およびクラスタ・エイリアスの使用をサポートしています。ネットワークは、クラスタをインターネット・エイリアスと呼ばれる、1つの名前を持った1つのシステムとして認識します。遠隔ホストは、クラスタ・エイリアスを使用してクラスタを1つのホストとしてアドレスを指定したり、1つのクラスタ・メンバのホスト名を使用して、クラスタ・メンバに個別にアドレスを指定することができます。

9.3.2 TCP/IP Services 管理ツールおよびユーティリティ

HP TCP/IP Services for OpenVMS は、100 を超える OpenVMS DCL スタイルのコマンドが含まれた、総合的な使いやすいネットワーク管理ツールを提供します。これらのコマンドは、TCP/IP> プロンプトで管理コマンドを実行することにより、TCP/IP Services の構成要素をローカルに構成し、監視し、チューニングすることができます。

UNIX 管理コマンドを使用して、コンポーネントの一部を管理することもできます。

使用しているシステムでの TCP/IP Services の管理方法の詳細については、『TCP/IP Services for OpenVMS Management』を参照してください。

9.4 TCP/IP ネットワークに参加するための準備

ホストで TCP/IP を構成するためには、その前に、一意な IP アドレスとホスト名が必要になります。ネットワーク・ハードウェア・アドレスは、ハードコード化されていて固定されていますが、IP アドレスはそれと異なり、ネットワーク管理者によって割り当てられます。ネットワークを公的なインターネットに接続する場合は、InterNIC から公式の一意なネットワーク ID を取得する必要があります。IP アドレスは 32 ビットの長さ (8 ビットが 4 つ) になっており、

これでネットワークとホストを識別します。ホスト名は、通信を簡単にする目的で、コンピュータに割り当てられた名前です。

IP アドレスは、ホスト ID とネットワーク ID を指定し、ホストがどのパケットを受信して、どのパケットを無視するか決定するための判断材料を提供します。ホストは、自身のアドレスおよびデスティネーション・コンピュータのアドレスを、サブネット・マスクと比較して、別のホストが同じサブネットにあるのか異なるサブネットにあるのか判断します。デスティネーションのネットワーク ID がソースのネットワーク ID と一致する場合、パケットはローカル・ネットワークのデスティネーション・ホストに配布されます。2 つのネットワーク ID が一致しない場合、パケットは IP ルータを経由してデスティネーション・コンピュータ宛に転送されます。

IP アドレスとホスト名の他に、次の情報をネットワーク管理者から聞いておく必要があります。

- システムがローカル・ネットワーク上にない TCP/IP ホストと通信する必要がある場合、省略時ゲートウェイのアドレス
- ネットワークが使用するルーティング・プロトコル
- ホスト名を IP アドレスに変換するドメイン・ネーム・サーバのアドレス
- ネットワーク・サブネットとブロードキャスト・マスク

また、どのエンド・ユーザ・サービスを提供するかということと、ユーザのシステムがクライアントとして動作するか、サーバとして動作するか、あるいは両方として動作するかについても決定しておく必要があります。

HP TCP/IP Services for OpenVMS のインストールおよび構成の計画の詳細については『TCP/IP Services for OpenVMS Concepts and Planning』を参照してください。

9.5 TCP/IP Services のインストールおよび構成

HP TCP/IP Services for OpenVMS ソフトウェアは、次のいずれかの方法でシステムにインストールすることができます。

- OpenVMS インストール・プロシージャのメニューから、オペレーティング・システムのアップグレードまたはインストールとして
- DCL コマンド PRODUCT INSTALL を使用し、レイヤード・アプリケーションとして

TCP/IP Services ソフトウェアが正常にインストールされたら、次のようにメニュー方式の構成プロシージャを起動し、使用しているシステムおよびネットワーク固有の特性に従って、ソフトウェアを構成します。

```
§ @SYS$MANAGER:TCPIP$CONFIG
```

インストールおよび構成の詳細については、『TCP/IP Services for OpenVMS インストール/コンフィギュレーション』を参照してください。

9.6 TCP/IP Services の開始と停止

HP TCP/IP Services for OpenVMS の標準ソフトウェアおよびオプションの構成要素を構成した後でこれを開始したり、所定のシャットダウンのために停止したりするには、次のコマンドを実行します。

```
§ @SYS$MANAGER:TCPIP$CONFIG
```

その後、TCP/IP Services for OpenVMS を開始または停止するためのメニュー・オプションを入力します。

9.7 TCP/IP Services のドキュメント

表 9-3 「HP TCP/IP Services for OpenVMS のドキュメント」は、HP TCP/IP Services for OpenVMS のためのドキュメントのリストです。TCP/IP Services for OpenVMS の計画、イン

ストール、構成、使用、および管理の詳細については、これらのドキュメントを参照してください。

表 9-3 HP TCP/IP Services for OpenVMS のドキュメント

マニュアル	説明
『TCP/IP Services for OpenVMS リリース・ノート』	インストール、アップグレード、互換などソフトウェアの変更点に関する情報について説明しています。また、新旧のソフトウェアの問題、制限事項、さらにソフトウェアとドキュメントの訂正事項についても説明しています。
『TCP/IP Services for OpenVMS Concepts and Planning』	TCP/IP の概念と構成要素について紹介し、ソフトウェア構成の計画を立てる上で役に立つ情報を提供しています。
『TCP/IP Services for OpenVMS インストール/コンフィギュレーション』	使用している OpenVMS ホストに HP TCP/IP Services for OpenVMS 製品をインストールし、構成する方法について説明しています。
『TCP/IP Services for OpenVMS User's Guide』	遠隔ファイル操作、電子メール、TELNET、TN3270、ネットワーク・プリントなど、TCP/IP Services で使用可能なアプリケーションの使用方法について説明しています。さらに、これらのサービスを使用して、プライベート・インターネットまたはワールドワイド・インターネット上にあるシステムと通信する方法についても説明しています。
『TCP/IP Services for OpenVMS Management』	HP TCP/IP Services for OpenVMS ソフトウェア製品の日常的な管理について説明しています。
『TCP/IP Services for OpenVMS Management Command Reference』	HP TCP/IP Services for OpenVMS コマンドについて説明しています。『TCP/IP Services for OpenVMS Management』の補完的ガイドです。
『TCP/IP Services for OpenVMS Tuning and Troubleshooting』	HP TCP/IP Services for OpenVMS のトラブルシューティングの方法およびその性能チューニングの方法について説明しています。
『TCP/IP Services for OpenVMS Guide to IPv6』	HP TCP/IP Services for OpenVMS IPv6 の機能と、使用しているシステムに IPv6 をインストールし、構成する方法について説明しています。
『HP TCP/IP Services for OpenVMS Sockets API and System Services Programming』	HP TCP/IP Services for OpenVMS を使用し、Berkeley Sockets Sockets または OpenVMS システム・サービスを使用してネットワーク・アプリケーションを開発する方法について説明しています。
『HP TCP/IP Services for OpenVMS ONC RPC Programming』	Open Network Computing Remote Call (ONC RPC) を使用した上位レベルのプログラミングの概要を示し、RPCGEN プロトコル・コンパイラを使用してアプリケーションを作成する方法および RPC プログラミング・インタフェースについて説明しています。
『HP TCP/IP Services for OpenVMS SNMP Programming and Reference』	Simple Network Management Protocol (SNMP) および SNMP アプリケーション・プログラミング・インタフェース (eSNMP) について説明しています。TCP/IP Services で提供されるサブエージェント、サブエージェントを管理するために提供されるユーティリティ、および独自にサブエージェントを構築する方法について説明しています。
『HP TCP/IP Services for OpenVMS Guide to SSH』	SSH for OpenVMS ソフトウェアを構成、設定、使用、および管理する方法を説明しています。

9.8 DECnet-Plus for OpenVMS ネットワーク・ソフトウェア

DECnet-Plus for OpenVMS を使用すると、さまざまなコンパックのオペレーティング・システムが、弊社のオペレーティング・システムや他社製のシステムと通信できるようになります。DECnet-Plus ネットワークは、遠隔システムとの通信、リソースの共有、分散処理などをサポートします。ネットワーク・ユーザは、ネットワーク上のすべてのシステム上にあるリソースに

アクセスできるようになります。ネットワークに参加しているそれぞれのシステムは、ネットワーク・ノードと呼ばれます。

DECnet-Plus は、DIGITAL ネットワーク・アーキテクチャ (DNA) の第 5 フェーズのインプリメントです。DNA Phase V では、OSI プロトコルが DECnet プロトコルに統合されています。さらに、DECnet-Plus は、インターネット規格 RFC 1006 とインターネットの草稿 RFC 1859 をサポートしています。これによって OSI アプリケーションと DECnet アプリケーションが TCP/IP を介して動作するようになります。このように DECnet-Plus を使用することによって、弊社のシステムからでも他社製のシステムからでも、アプリケーションは、任意の DECnet Phase IV ベースのシステムまたは OSI ベースのシステム上の OSI および DECnet アプリケーションと通信できるようになります。

DECnet-Plus の DECnet (Phase IV) との主な違いは、一般的なプロトコルをサポートしているという点にあります。

DECnet-Plus では、ネットワーク機能を拡張するためのさまざまな機能が用意されています。これらの機能を次に示します。

- グローバルな名前とディレクトリ・サービス。大規模なネットワークで、数百万(理論上)のネットワーク・オブジェクトに対応するアドレッシング情報について、格納、管理、およびアクセスを簡単に実行できるようになる。このオブジェクトには、エンド・システム、ユーザ、プリンタ、ファイル、ディスクなどがある。
- オプションのローカル名およびディレクトリ・サービス。グローバルなディレクトリ・サービスを使用する必要がない、小さなネットワークの場合にオプションで使用できる。
- Network Control Language (NCL) を使用した拡張ネットワーク管理機能。
- ホスト・ベースのルーティング。OpenVMS システムが、ルーティング・ドメイン内で DECnet-Plus の中間システムとして動作できるようになる。

LAN から WAN ヘルパーリングする必要があるが、専用ルータを使用しないで既存のシステムでルーティングする構成の場合、この機能が特に便利になる。ホスト・ベースのルーティングは、高いスループットが要求されるネットワーク構成には向いていない。

- OSI 規格のアドレス形式の場合のアドレッシング機能の向上。これにより、実質的に無限の数のノードで、一意なアドレッシングが使用できるようになる。既存の Phase IV アドレスも、DECnet-Plus にアップグレードしたシステムで使用し続けることができる。この場合 Phase IV アドレスは、構成プロシージャにより、OSI アドレス形式に自動的に変換される (DECnet Phase IV 互換アドレスと同様)。
- アドレスの自動構成。隣接するルータが、ローカル・ノードのノード・アドレスを構成できるようになる。
- OSI コンポーネントの単一の構成。

DECnet-Plus ソフトウェアには、DECnet-Plus ソフトウェアと同様、(VAX システム上の) X.25、ワイド・エリア・デバイス・ドライバ (WANDD)、ファイル転送、アクセス、管理 (FTAM)、および仮想ターミナル (VT) の各アプリケーションが含まれる。Alpha システムや I64 システムでは、X.25 のサポートが、DECnet-Plus ソフトウェアと切り離されている。

9.8.1 DECnet-Plus のノード名

DECnet ノード名の命名規則は、次の 2 種類の DECnet 機能に対応しています。

- DECnet-Plus の完全な名前

完全な名前は階層構造になった DECnet ノード名で、DECdns ネーム・サービスに格納できる。完全な名前は、最大で 255 文字の長さにする事ができる。

- DECnet-Plus でノード同意語と呼ばれる DECnet Phase IV のノード名

これらの名前は、DECnet Phase IV で使用される短い名前で、6 文字以下という制限を受ける。これらの名前を使用すると、DECnet-Plus が同じネットワーク内の DECnet Phase IV システムと下位互換性を持つようになる。

完全な名前の構文

完全な名前では、一般的に次のような形式が使用されます。

ネームスペース:.ディレクトリディレクトリ.ノード名

ネームスペース	グローバル・ネーム・サービスを指定
ディレクトリディレクトリ	ネーム・サービス内の階層化ディレクトリ・パスを定義
ノード名	DECnet ノードを定義する特定のオブジェクト

次に、ローカル・ネームスペース、DECdns、DNS/BIND に対応する完全なノード名の例を示します。

ローカル・ネームスペース:	LOCAL:.CPlace
DECdns:	ACME:.warren.CPlace
Domain:	CPlace.warren.acme.com

ユーザが入力した完全な名前は、大文字と小文字が区別された状態でシステムに格納されます。ただしエントリと、格納された名前をマッチングさせる時には、システムは大文字と小文字を区別しません。つまりユーザが Acme と入力した場合でも、システムは、それを ACME と認識します。

完全な名前についての詳細は、DECnet-Plus のドキュメントを参照してください。

9.8.2 OpenVMS Cluster システムのサポート

DECnet-Plus ソフトウェアは、OpenVMS Cluster システムをサポートし、OpenVMS Cluster の別名の使用についてもサポートしています。DECnet-Plus では、それぞれの OpenVMS Cluster に対して別名を 3 つ使用できるようになっています。DECnet Phase IV の場合、ノードを DECnet-Plus の別名のメンバにすることはできません。それぞれの別名を、DECnet Phase IV のノードで使用できるよう構成する必要があります。

CLUSTER_CONFIG.COM コマンド・プロシージャは、OpenVMS Cluster の構成を実行します。このとき任意のクラスタ・メンバから、クラスタのすべてのメンバについて構成することができます。これにより、DECnet-Plus for OpenVMS の NET\$CONFIGURE.COM コマンド・プロシージャが起動され、NCL 初期化スクリプトに対して必要な変更が行われます。OpenVMS Cluster を構成するには CLUSTER_CONFIG.COM を使用します。CLUSTER_CONFIG.COM がすでに使用されている場合に、DECnet-Plus のサテライト・ノードを構成するには、NET\$CONFIGURE.COM を直接使用します。

9.8.3 DECnet-Plus 管理ツールおよびユーティリティ

DECnet-Plus for OpenVMS で提供されるツールにより、次の作業ができるようになります。

- ローカルおよび遠隔の DECnet Phase V 構成要素を管理する。Network Control Language (NCL) コマンド行インタフェースと Motif ベースのウィンドウ・インタフェース (NET\$MGMT) の 2 つのインタフェースが使用できる。

DECNET_MIGRATE ツールが提供されているため、DECnet Phase IV のそれぞれの NCP コマンドを NCL コマンドに、またはコマンド・プロシージャ内の NCP コマンドを NCL コマンドに変更することができるようになっている。DECnet-Plus を使用したことはないが NCL については熟知しており、慣れている NCP コマンドを NCL 構文で指定したい場合は、DECNET_MIGRATE を使用できる。

- NCP Emulator (NCP.EXE) で遠隔の DECnet Phase IV ノードを管理する。このユーティリティは、NCP コマンドの大部分をサポートしている。ただし NCL に代わって、DECnet-Plus システムの管理を行うためのものではない。
- DECnet-Plus for OpenVMS 初期化スクリプト (SYS\$MANAGER:NET\$*.NCL 形式のファイル) を使用する。

- (MOPを使用して) ダウンライン・ロード，アップライン・ロード，遠隔コンソール接続，ループバック・テスト・サポートなどの保守操作を実行する。DECnet-Plus for OpenVMS では，同時ダウンライン・ロードの拡張サポートとパフォーマンスが提供されている。MOP，およびこのプロセスを開始する方法については，『DECnet-Plus for OpenVMS Network Management guide』を参照。
- EVD を使用して，拡張イベント・ロギングを実行する。
- トラブルシューティングのために Common Trace Facility (CTF) を使用する。
- DECNET_REGISTER ツールを使用して，ローカル・ネームスペースおよび DECdns ネームスペースに対応する，ネットワーク内のノード名を管理する。

9.9 DECnet-Plus ネットワークに参加するための準備

DECnet-Plus ノードを構成する前に，アドレッシング，ネーム・サービスの使用，タイム・サービス，ルータについて決定しておく必要があります。また X.25 ソフトウェアに固有のライセンスの従属関係についても注意する必要があります。

準備の詳細については，『DECnet-Plus Planning Guide』を参照してください。

9.10 DECnet-Plus のインストールおよび構成

DECnet-Plus for OpenVMS ソフトウェアは，次のいずれかの方法でシステムにインストールすることができます。

- OpenVMS インストール・プロシージャのメニューから，オペレーティング・システムのアップグレードまたはインストールとして
- DCL コマンド PRODUCT INSTALL を使用し，レイヤード・アプリケーションとして

DECnet-Plus ソフトウェアが正常にインストールされたら，次のようにメニュー方式の構成プロシージャを起動し，使用しているシステムおよびネットワーク固有の特性に従って，ソフトウェアを構成します。

- DECnet Phase IV ノードからアップグレードしており，既存の Phase IV 構成を使用するよう計画し，使用しているノードが OpenVMS Cluster の一部でない場合には，Fast 構成オプションを使用する。次のコマンドを入力する。

```
$ @SYS$MANAGER:NET$CONFIGURE
```

- ノードがクラスタ内にあり，DECnet-Plus をアップグレードまたは再構成しており，DECnet を TCP/IP を介して使用する場合には，Basic 構成オプションを使用する。次のコマンドを入力する。

```
$ @SYS$MANAGER:NET$CONFIGURE BASIC
```

- ノードの構成が複雑で，これをカスタマイズする必要がある場合には，Advanced 構成オプションを使用する。次のコマンドを入力する。

```
$ @SYS$MANAGER:NET$CONFIGURE ADVANCED
```

インストールおよび構成の詳細については，『DECnet-Plus for OpenVMS Installation and Basic Configuration』を参照してください。

9.11 DECnet over TCP/IP の使用方法

DECnet over TCP/IP 機能により，DECnet Phase V アーキテクチャが，TCP/IP ネットワークと共存し，通信することができるように拡張されます。この機能を使用するには，有効な DECnet ライセンスと，PATHWORKS Internet Protocol (PWIP) インタフェースをサポートする，ライセンスを受けインストールされている TCP/IP 製品が必要です。

次のような目的がある場合には、DECnet over TCP/IP を使用する必要があります。

- DECnet から TCP/IP への変更。業務上の必要から、TCP/IP ベースのネットワークに徐々に移行するため。DECnet over TCP/IP の機能は、ユーザにとって透過的に設計されている。
- 共存。DECnet アプリケーションまたは OSI アプリケーション、あるいはその両方を実行するため。

DECnet over TCP/IP を使用すると、次の作業ができるようになります。

- TCP/IP ネットワーク・バックボーンを介して DECnet ネットワークを拡張する。DECnet over TCP/IP を使用すると、複数のネーム・サービス (DECdns, 大規模な LOCAL ファイル, および DNS/BIND) を組み合わせることができる。
- 既存の 2 つの DECnet ネットワークを結合することにより、ノード番号を再設定することなくネットワークを拡張する。
- バックボーンの一部または全体で、IP 専用トラフィックを使用する。DECnet over TCP/IP を使用すると、DECnet アプリケーションまたは OSI アプリケーションは、TCP/IP プロトコル・スタックおよび IP ネットワーク・バックボーンのより下位のレベルで実行することができる。
- 機能を、ノード間ベースまたはネットワーク全体で使用可能にする。

DECnet over TCP/IP 機能は、TCP/IP を使用して複数の DECnet ノード間に論理リンクを形成します。PATHWORKS IP Driver を使用して、TCP とインタフェースを取ります。DECnet アプリケーションは、TCP/IP によって接続された DECnet ノード間で透過的に実行されます。DECnet ノードのユーザは、DECnet ノード同意語または IP フルネームを使用して、相互に接続することができます。次に例を示します。

```
$ SET HOST SYSABC
$ SET HOST SYSABC.boston.acme.com
$ SET HOST 16.12.42.19
```

使用しているシステムで DECnet over TCP/IP を使用可能にするには、その前に次の作業を実行しなければなりません。

- システムに DECnet-Plus をインストールし、構成する。
高度なオプションを使用して NET\$CONFIGURE プロシージャを実行するときには、ディレクトリ・サービスのリストに、少なくとも Domain ディレクトリ・サービスを指定しなければなりません。また、「Configure the OSI transport or run over TCP/IP?」という質問に対して YES と答える必要があります。
- システムに TCP/IP Services をインストールし、構成する。
TCP/IP\$CONFIG 構成プロシージャを実行する場合には、PWIP ドライバを使用可能にし、DECnet-Plus と TCP/IP Services の間にブリッジが形成されるようにする。PWIP ドライバは、Optional Components メニューの Option 1 にリストされています。

DECnet over TCP/IP を使用可能にしたり、使用したりする方法の詳細については、『DECnet-Plus for OpenVMS Applications Installation and Advanced Configuration』および『DECnet-Plus for OpenVMS Network Management』を参照してください。

9.12 DECnet Phase IV から DECnet-Plus への移行

ネットワークを DECnet Phase IV から DECnet-Plus に移行するときには、ネットワークを DECnet-Plus に部分的に移行することも、ネットワーク全体を移行することも可能です。DECnet-Plus には下位互換性があるため、使用しているシステムとネットワークを、DECnet Phase IV アプリケーションやルーティングなどを使用してこれまでどおり実行するように選択することができます。使用可能な追加機能は、準備ができればいつでも、DECnet-Plus からインプリメントすることができます。変更点のほとんどは、ネットワーク環境に関係しています。これらは、ほとんど全体が、ユーザおよびアプリケーションに対して透過的です。

豊富な自動化ツール (DECnet 移行ユーティリティおよび NCP Emulator) の他、単純化された構成プロセスも使用でき、全機能をインプリメントした DECnet-Plus への移行しやすくするために役立ちます。

ネットワークの移行の詳細については、『DECnet-Plus Planning Guide』を参照してください。

9.13 DECnet-Plus の開始と停止

DECnet-Plus を OpenVMS インストール・メニューからインストールすると、DECnet-Plus ソフトウェアは自動的に開始します。何らかの理由(たとえば SYS\$STARTUP:NET\$SHUTDOWN.COM を実行してネットワークをシャット・ダウンした後であるなど)で DECnet-Plus を再始動する必要がある場合には、次のコマンドを実行します。

```
$ @SYS$STARTUP:NET$STARTUP
```

DECnet-Plus ソフトウェアをシャットダウンすると、システム上のさまざまなネットワーク構成要素が使用不可能になり削除される場合には、次のコマンドを入力します。

```
$ @SYS$MANAGER:NET$SHUTDOWN
```

9.14 DECnet-Plus Documentation

表 9-4 「DECnet-Plus for OpenVMS のドキュメント」は、DECnet-Plus for OpenVMS ソフトウェアをサポートするドキュメントのリストです。DECnet-Plus の計画、インストール、構成、および管理の詳細についてはこれらのドキュメントを参照してください。

表 9-4 DECnet-Plus for OpenVMS のドキュメント

マニュアル	説明
『DECnet-Plus for OpenVMS Release Notes』	ソフトウェアの変更、インストール、アップグレード、互換性などの情報、新旧のソフトウェアの問題、制限事項などについて解説しています。またソフトウェアとドキュメントの訂正事項についても説明しています。このテキスト・ファイルは、構成プロセスでプリントすることができます。
『DECnet-Plus for OpenVMS Introduction and User's Guide』	システムのネットワークについて紹介し、ユーザ情報について解説しています。
『DECnet-Plus for OpenVMS Installation and Basic Configuration』	DECnet-Plus のインストール方法、BASIC 構成オプションの実行方法について説明しています。
『DECnet-Plus for OpenVMS Applications Installation and Advanced Configuration』	ネットワーク・アプリケーションのインストール方法と構成方法について説明し、ADVANCED 構成オプションの実行方法について説明しています。
『DECnet-Plus for OpenVMS Installation Quick Reference Card』	インストール時にシステムを DECnet-Plus にアップグレードする場合に使用できる簡単なリファレンスが提供されます。
『DECnet-Plus Planning Guide』	DECnet Phase IV の機能を DECnet Phase V に移行する手順を紹介しています。
『DECnet-Plus for OpenVMS Network Management』	DECnet-Plus システムのネットワーク管理の概念とタスクについて解説しています。
『DECnet-Plus Network Control Language Reference』	すべての NCL コマンドの解説と例を紹介しています。
『DECnet-Plus Problem Solving』	ネットワークが動作しているときに発生する DECnet-Plus の問題を特定し解決する方法について説明しています。またループバック・テストを実行する方法についても説明しています。

第10章 ローカル・エリア・ネットワーク (LAN) ソフトウェアの管理

この章では、LAN ソフトウェアの動作について説明するとともに、LAN ソフトウェアを管理するためにシステム上で実行する作業について説明します。

この章の内容

この章では次の作業について説明します。

作業	参照箇所
LANACP LAN サーバ・プロセスの実行	10.3.1 項 「LANACP LAN サーバ・プロセスの実行」
LANCP の起動と実行	10.4.1 項 「LANCP の起動と実行」
LAN デバイスの管理	10.5 項 「LAN デバイスの管理」
LAN デバイス・データベースの管理	10.6 項 「LAN デバイス・データベースの管理」
LAN ノード・データベースの管理	10.7 項 「LAN ノード・データベースの管理」
DECnet MOP から LAN MOP への移行	10.8.2 項 「DECnet MOP から LAN MOP への移行」
CLUSTER_CONFIG_LAN.COM および LAN MOP	10.8.3 項 「CLUSTER_CONFIG_LAN.COM と LAN MOP」
MOP ダウンライン・ロード・サービスの管理	10.9 項 「LAN MOP ダウンライン・ロード・サービスの管理」
MOP コンソール・キャリアの始動	10.9.8 項 「MOP コンソール・キャリア」
MOP トリガ・ブートの要求	10.9.9 項 「MOP トリガ・ブート」
LAN フェイルオーバの使用	10.10 項 「LAN フェイルオーバについて」

さらに、次の項目について説明します。

項目	参照箇所
ローカル・エリア・ネットワーク	10.1 項 「ローカル・エリア・ネットワーク」
LANACP LAN サーバ・プロセス	10.3 項 「LANACP LAN サーバ・プロセス」
LANCP ユーティリティ	10.4 項 「LANCP ユーティリティ」
MOP ダウンライン・ロード・サービス	10.8 項 「LAN MOP」
LAN フェイルオーバ	10.10 項 「LAN フェイルオーバについて」

10.1 ローカル・エリア・ネットワーク

ローカル・エリア・ネットワーク (LAN) は、部屋、建物、あるいは建物群 (たとえば大学) のような限られた範囲内において、情報処理デバイスを接続するための、通信チャネルを提供します。LAN 内のノードは、次のタイプのデータ転送媒体によってリンクすることができます。

- イーサネット

最も初期に開発された一般的な LAN のひとつ。イーサネットは、一般的な LAN アプリケーション (たとえば イーサネット・アドレス)、または Intel®, Xerox, Digital による企業間のイーサネット仕様をインプリメントする固有の CSMA/CD (衝突検出機能付きキャリア検知多重アクセス) テクノロジーを指す。

イーサネット LAN には、次の 3 種類がある。

- 転送速度 10Mbps のイーサネット (IEEE 802.3)
- 転送速度 100Mbps の高速イーサネット (IEEE 802.3u)
- 転送速度 1000Mbps のギガビット・イーサネット (IEEE 802.3z)

この 3 種類のイーサネットは、どれも、CSMA/CD プロトコル、同一のフレーム形式、同一のフレーム・サイズを用いている。ギガビット・イーサネットの場合は、ジャンボ・フレームという大きなサイズのフレームをオプションで使用できる。

- FDDI (Fiber Distributed Data Interface)
デュアルリングのトークン・リング LAN としてインプリメントされている。
- トークン・リング
IEEE 802.5 のトークン・パッシング式リング。
- ATM (非同期転送モード)
次の標準をサポートする。
 - ATM を介した LAN エミュレーションは ATM Forum の LAN Emulation V1.0 (LANE) 標準をサポート。
 - ATM を介した Classical IP は RFC 1577 標準をサポート (DGLTA, DGLPA, DGLPB のみ)。
- 共用メモリ
共用メモリを使用した Galaxy ノード間でのイーサネットのエミュレーション。

10.1.1 LAN の特性

LAN コントローラは、追加の外部ハードウェアとともに、イーサネット、FDDI、トークン・リング、ATM または Classical IP (RFC 1577) を介した LAN エミュレーションの仕様をインプリメントするデバイスです。LAN コントローラとローカル・システムでノードを構成します。LAN コントローラはシステム・バスを介してローカル・システムと通信します。また通信媒体を介して、イーサネット、FDDI、トークン・リング、または ATM を介した LAN エミュレーションの仕様をインプリメントする遠隔システムと通信します。イーサネットの仕様については『The Ethernet-Data Link Layer and Physical Layer Specification』と、IEEE から入手可能な仕様書に記述されています。FDDI の仕様は ANSI から入手可能です。トークン・リングの仕様は IEEE から入手可能です。ATM を介した LAN エミュレーションの仕様は ATM Forum から入手可能です。

アプリケーション・プログラムは、LAN ドライバの QIO インタフェースおよび VCI インタフェースを使って、LAN 上の他のノードとの間で入出力処理を実行します。QIO インタフェースおよび VCI インタフェースについての詳細は、『HP OpenVMS I/O User's Reference Manual』を参照してください。

表 10-1 「LAN 媒体の特性」に、LAN 媒体のタイプの相違点を要約します。

表 10-1 LAN 媒体の特性

媒体	速度	最大フレーム・サイズ	最大ケーブル長 ¹
イーサネット 802.3	10Mbps	1518 バイト	10Base-T - 100 m, 10Base-2 - 185 m, 10Base-5 - 500 m
高速イーサネット 802.3u	100Mbps	1518 バイト	100Base-TX - 100 m 100Base-FX - 2 Km
ギガビット・イーサネット 802.3z	1000Mbps	1518 または 9018 バイト	1000Base-SX 光ファイバ - 550 m, 1000BaseT 銅線 UTP - 100 m
FDDI	100Mbps	4495 バイト	UTP - 100 m, マルチノード・ファイバ - 2 Km

表 10-1 LAN 媒体の特性 (続き)

媒体	速度	最大フレーム・サイズ	最大ケーブル長 ¹
トークン・リング	802.5 4 または 16 Mbps	4462 バイト	UTP - 45 m, STP - 110 m
ATM を介した LAN エミュレーション	155Mbps または 622Mbps	1516, 4544, または 9234	マルチノード・ファイバ - 2 Km, UTP - 300 m

1 大規模ネットワークは、ハブ、ブリッジ、スイッチによって構築できる。

10.1.1.1 イーサネット LAN

イーサネット・ネットワークは、各種のケーブルで接続され、CSMA/CD プロトコルに従って通信を行うノード群からなっています。最も簡単な形式のネットワークでは、2つのシステムが、ポイント・ツー・ポイント形式でケーブル 1 本で接続されます。より複雑な構成では、より多くのシステムを接続するために、イーサネット・スイッチやハブが追加されます。

10.1.1.2 FDDI LAN

FDDI はツリー・トポロジのデュアル・リングを使用します。一方のリングを 1 次リング、他方をバックアップとして使用し、柔軟性、管理の容易さ、可用性を高めるために、ツリー構造を採用しています。

FDDI ネットワークとイーサネット・ネットワークを組み合わせると、1つの拡張 LAN を形成できます。こうすることにより、FDDI に接続されたシステム上で実行されるアプリケーションを、イーサネットに接続されたシステムで実行されるアプリケーションに接続できます。

FDDI 集信デバイスまたはスイッチは、VAX や Alpha ノード、または FDDI とイーサネット間のブリッジのような FDDI デバイスを、LAN に接続することができます。

10.1.1.3 トークン・リング LAN

トークン・リング・コントローラは、シールドまたは非シールドのツイストペア・ワイヤを使って、リングにアクセスします。ブリッジで直接接続されたトークン・リング LAN を他のタイプの LAN に接続するのは難しいので、注意してください。ただし、他の LAN へのルーティング・プロトコルにより相互接続が可能です。

10.1.1.4 ATM LAN

ATM を介した LAN エミュレーションは、接続に基づいた光ファイバ・ネットワークから構成されます。OpenVMS ATM ネットワークは、データ転送には AAL5 ATM 適応層を使用します。

ATM を介した LAN エミュレーションについては、OpenVMS では LAN エミュレーション・クライアント (LEC) だけをインプリメントしており、LAN エミュレーション・サーバ (LES), Broadcast and Unknown (BUS), または LAN エミュレーション構成サーバ (LECS) はインプリメントしていません。LES, BUS, および LECS は、ATM スイッチなど他のファシリティにより提供されなければなりません。OpenVMS がサポートしているのは、ATM アダプタ 1 つにつき 8 つの LAN エミュレーション・クライアントです。

ATM を介した Classical IP (CLIP) は、イーサネット・インタフェース (802.3) と同じ意味を持つデータリンク・レベル・デバイス・インプリメントします。このイーサネット・インタフェースは、ATM ネットワークを介して 802.3 (IEEE イーサネット) フレームを送信するとき TCP/IP プロトコルにより使用されます。OpenVMS が ATM を介して IP データグラムを交換する場合、RFC1577 (ATM を介した Classical IP) に基づいたモデルに従います。

10.1.2 LAN アドレス

LAN 上のノードは、一意のアドレスで識別されます。使用するアドレスによって、メッセージを LAN 上の 1 つ、数個、またはすべてのノードに同時に送信することができます。

適用の際、IEEE はアドレス・ブロックを LAN ノードの製作者に割り当てます。したがって、どのメーカーも固有のアドレス・セットを持っています。通常、割り当てられた物理アドレス・ブロックのうちの 1 アドレスが、永久に各コントローラ (通常は読み込み専用メモリ) に対応付けられます。このアドレスは、コントローラのハードウェア・アドレス (MAC アドレス) と呼ばれます。各コントローラには固有のハードウェア・アドレスがあります。

LAN アドレスは長さが 48 ビットです。LAN アドレスは、6 対の 16 進数字 (6 バイト) をハイフンで区切って表現します (たとえば AA-01-23-45-67-FF)。バイトは転送される順に、左から右に表示されます。各バイト中のビットは右から左に転送されます。この例では、バイト AA が最初に転送され、バイト FF が最後に転送されます。

LAN アドレスは、アドレスの最初のバイトの最下位ビット (このビットが最初に転送される) の値によって、単一ノードのアドレス、またはマルチキャスト・アドレスになります。ノード・アドレスの 2 つのタイプは次のとおりです。

- 個別アドレス (物理アドレス)—LAN 上の単一ノードの固有のアドレス。物理アドレスの第 1 バイトの最下位ビットは 0 (たとえば、物理アドレス AA-00-03-00-FC-00 では、バイト AA は 2 進数で 1010 1010 であり、最下位ビットの値は 0)。
- マルチキャスト・アドレス — ある LAN 上の 1 つまたは複数のノードが認識するグループ・アドレス。マルチキャスト・アドレスの第 1 バイトの最下位ビットは 1 (たとえば、マルチキャスト・アドレス 0B-22-22-22-22-22 では、バイト 0B は 2 進数で 0000 1011 であり、最下位ビットの値は 1)。

トークン・リング・デバイスは、IEEE 802 規格のマルチキャスト・アドレスをサポートせず、機能アドレスをサポートする。機能アドレスとは、ローカルに管理されるグループ・アドレスであり、31 の値が可能。各機能アドレスは、アドレスの 3 番目から 6 番目のバイトに 1 ビットを設定し、バイト 1 と 2 は 03-00 (ビットの逆形式では C0:00) である。マルチキャスト・アドレスを機能アドレスに変換するには、LANCP の SET DEVICE/MAP コマンドを使用する。

10.2 ローカル・エリア・ネットワークの管理

ローカル・エリア・ネットワーク (LAN) ソフトウェアには、OpenVMS LAN ドライバ・システム・ソフトウェアとともに動作する次の 2 つのシステム管理ツールが含まれています。

- LANCP (ローカル・エリア・ネットワーク制御プログラム)
- LANACP LAN サーバ・プロセス

LAN システム管理ツールは次のことを行います。

- システム管理者が LAN パラメータを設定して、LAN 環境をカスタマイズできるようにする。
- LAN の設定値とカウンタを表示する。
- ターミナル・サーバや X ターミナル、LAN ベースのプリンタなどのデバイス、および OpenVMS Cluster 環境でのサテライトのブート用に保守操作プロトコル (MOP) ダウンライン・ロード・サポートを提供する。
- MOP コンソール・キャリアと MOP トリガ・ブートをサポートする。

表 10-2 「LAN システム管理の強化」は、LAN 管理ソフトウェアと、OpenVMS Alpha、OpenVMS VAX、および OpenVMS I64 を実行するシステムでサポートされる機能についての説明です。

表 10-2 LAN システム管理の強化

ユーティリティ	説明	OpenVMS のサポート
LANACP (LAN 補助制御プログラム)	MOP ダウンライン・ロード・サーバ・プロセスの提供を主な機能とするサーバ・プロセスとして実行される。LAN 運用時デバイス・データベースおよび LAN 運用時ノード・データベースの保守も行う。	LANACP ユーティリティは、OpenVMS バージョン 7.0 以降が動作している VAX システムおよび Alpha システムと、OpenVMS バージョン 8.2 以降が動作している I64 システムで同等の機能を提供する。
LANCP (LAN 制御プログラム)	システム管理者が LAN ソフトウェアのパラメータを制御して、LAN ソフトウェアから情報を得られるようにする。LANCP ユーティリティで次のことができる。 <ul style="list-style-type: none"> LAN デバイスのカウンタ、リビジョン、構成情報を取得する。 システム上で LAN デバイスの操作パラメータを変更する。 パーマメントおよび運用時 LAN デバイス・データベースおよびノード・データベースを保守する。 LANACP LAN サーバ・プロセス (MOP ダウンライン・ロード・サーバ関連機能を含む) を制御する。 MOP コンソール・キャリア接続を開始する。 MOP トリガ・ブート要求を他のノードに送信する。 	<p>OpenVMS Alpha バージョン 6.1 には初期にインプリメントされた LANCP は含まれるが、MOP 関連機能は含まれない。</p> <p>OpenVMS バージョン 6.2 (VAX および Alpha) には、MOP 関連機能が含まれ、その機能の一部を VAX システムに対して拡張している。VAX システム、Alpha システム、および I64 システム上でサポートされている LAN ユーティリティ機能を以下に示す。</p> <p>OpenVMS Alpha Version 7.3-1 およびそれ以降:</p> <ul style="list-style-type: none"> - LAN デバイスの操作パラメータの変更? Yes - LAN デバイス情報の表示? Yes - MOP 機能のサポート? Yes <p>OpenVMS VAX Version 7.3 およびそれ以降:</p> <ul style="list-style-type: none"> - LAN デバイスの操作パラメータの変更? No - LAN デバイス情報の表示? 制限あり - MOP 機能のサポート? Yes <p>OpenVMS I64 Version 8.2 およびそれ以降:</p> <ul style="list-style-type: none"> - LAN デバイスの操作パラメータの変更? Yes - LAN デバイス情報の表示? Yes - MOP 機能のサポート? Yes

10.3 LANACP LAN サーバ・プロセス

LANACP LAN サーバ・プロセスを実行して、次の処理を行うことができます。

- LAN 運用時ノード・データベースの保守
- LAN 運用時デバイス・データベースの保守
- MOP ダウンライン・ロード

LANCP ユーティリティを使用すると、LANACP プロセスに対して命令を出すことができます。LANACP に関連する主な 3 つのファイルは次のとおりです。

- SYS\$SYSTEM:LANACP.EXE
LANACP ユーティリティ・プログラム。
- SYS\$STARTUP:LAN\$STARTUP.COM
LANACP サーバ・プロセスを起動する。

- SYS\$STARTUP:VMS\$DEVICE_STARTUP.COM

このファイルは、システムのスタートアップ時に LANACP を自動的に起動できるエントリーを含む。

さらに、LANACP LAN サーバ・プロセスには関連する 4 つのシステム論理名があり、これについては表 10-3 「LANACP システム論理名」で説明します。

表 10-3 LANACP システム論理名

コンポーネント	説明
LAN\$DLL	ダウンライン・ロード・ファイルの位置を定義する。ファイル位置はロード要求では指定されず、LAN 運用時ノード・データベースにも明示的に提供されていない。省略時には SYS\$SYSROOT:[MOM\$SYSTEM] と定義される。
LAN\$NODE_DATABASE	LAN パーマネント・ノード・データベースの位置を定義する。省略時には SYS\$COMMON:[SYSEXE]LAN\$NODE_DATABASE.DAT と定義される。
LAN\$DEVICE_DATABASE	LAN パーマネント・デバイス・データベースの位置を定義する。省略時には SYS\$SPECIFIC:[SYSEXE]LAN\$DEVICE_DATABASE.DAT と定義される。
LAN\$ACP	LANACP LAN サーバ・プロセスのログ・ファイルの位置を定義する。このログ・ファイルには、LAN パーマネント・デバイス・データベースおよびノード・データベースの変更や、ロード要求、ロード状態情報を記述するエントリーが含まれる。省略時には SYS\$MANAGER:LAN\$ACP.LOG と定義される。

10.3.1 LANACP LAN サーバ・プロセスの実行

LANACP LAN サーバ・プロセスを開始するには、DCL プロンプトに対して @SYS\$STARTUP:LAN\$STARTUP とタイプします。

10.3.2 LANACP LAN サーバ・プロセスの終了

LANACP LAN サーバ・プロセスを終了するには、LANCP ユーティリティ・プロンプトで SET ACP/STOP コマンドを入力します。

10.4 LANCP ユーティリティ

LANCP ユーティリティでは、LAN パラメータを設定して、表示することができます。10.4.1 項「LANCP の起動と実行」で LANCP ユーティリティの起動方法について説明します。表 10-4 「LANCP ユーティリティの機能」は、LAN 機能の説明と、その機能の実行に役立つ LANCP コマンドについての参照箇所を示しています。

表 10-4 LANCP ユーティリティの機能

作業	参照箇所
LAN デバイスの管理	10.5 項 「LAN デバイスの管理」
LAN デバイス・データベースの管理	10.6 項 「LAN デバイス・データベースの管理」
LAN ノード・データベースの管理	10.7 項 「LAN ノード・データベースの管理」
MOP ダウンライン・ロード・サービスの管理	10.9 項 「LAN MOP ダウンライン・ロード・サービスの管理」
MOP コンソール・キャリア接続の起動	10.9.8 項 「MOP コンソール・キャリア」
MOP トリガ・ブート要求の送信	10.9.9 項 「MOP トリガ・ブート」

10.4.1 LANCP の起動と実行

表 10-5 「LANCP ユーティリティの起動」は、LANCP ユーティリティ (SYS\$SYSTEM:LANCP.EXE) の起動方法と実行方法について説明しています。

表 10-5 LANCP ユーティリティの起動

コマンド	例
RUN コマンドの使用	DCL コマンド・プロンプトに対して、次のように入力する。 <pre>\$ RUN SYS\$SYSTEM:LANCP</pre> これで LANCP ユーティリティから LANCP プロンプトが表示されるので、LANCP コマンドを入力できる。
LANCP をフォーリン・コマンドとして定義	DCL プロンプトに対して、またはスタートアップ・コマンド・ファイルかログイン・コマンド・ファイルに、次のように入力する。 <pre>\$ LANCP ::= \$SYS\$SYSTEM:LANCP</pre> これで、DCL プロンプトからコマンド LANCP を入力してユーティリティを起動し、LANCP コマンドを入力できるようになる。 LANCP コマンドを入力する場合には、次の点に注意する。 <ul style="list-style-type: none"> • コマンド修飾子を指定しなければ、LANCP ユーティリティが LANCP プロンプトを表示するので、それに対してコマンドを入力できる。 • コマンド修飾子を指定すると、LANCP ユーティリティはコマンドを実行した後終了し、DCL コマンド・プロンプトが表示される。

LANCP> プロンプトに対して LANCP コマンドを入力できます。

LANCP ユーティリティについての情報を得るには、LANCP> プロンプトで HELP コマンドを入力します。

LANCP ユーティリティを終了するには、LANCP> プロンプトで EXIT コマンドを入力するか、Ctrl/Z を押します。

10.4.2 LANCP コマンド

表 10-6 「LANCP コマンド」に、LANCP コマンドについての要約を示します。

表 10-6 LANCP コマンド

コマンド	機能
@ (実行プロシージャ)	コマンド・プロシージャを実行する。
CLEAR DEVICE	LAN 運用時デバイス・データベースからデバイス・データを削除する。
CLEAR DLL または MOPDLL	全ノードおよびデバイスで、MOP ダウンライン・ロード・カウンタをクリアする。
CLEAR NODE	LAN 運用時ノード・データベースからノードを削除する。
CONNECT NODE	MOP コンソール・キャリア・プロトコルを使用して管理インタフェースをインプリメントする、ターミナル・サーバなどの LAN デバイスに接続する。
CONVERT DEVICE_DATABASE	デバイス・データベースを LANCP で要求される現在の形式に変換する。
CONVERT NODE_DATABASE	ノード・データベースを LANCP で要求される現在の形式に変換する。
DEFINE DEVICE	デバイスを LAN パーマネント・デバイス・データベースに入力するか、既存のエントリを変更する。
DEFINE NODE	ノードを LAN パーマネント・ノード・データベースに入力するか、既存のエントリを変更する。
EXIT	LANCP の実行を終了し、制御を DCL コマンド・レベルに戻す。
HELP	LANCP ユーティリティに関するオンライン・ヘルプを提供する。
LIST DEVICE	LAN パーマネント・デバイス・データベースの情報を表示する。
LIST NODE	LAN パーマネント・ノード・データベースの情報を表示する。

表 10-6 LANCP コマンド (続き)

コマンド	機能
PURGE DEVICE	デバイス・データを LAN パーマネント・デバイス・データベースから削除する。
PURGE NODE	ノードを LAN パーマネント・ノード・データベースから削除する。
SET ACP	LANACP LAN サーバ・プロセスの操作を変更する。
SET DEVICE	LAN 運用時デバイス・データベースとデバイス自体にあるデバイス特性を変更する。
SET NODE	ノードを LAN 運用時ノード・データベースに入力するか、既存のエントリを変更する。
SHOW ACP	現在設定されている LANCP オプションと LANACP オプションを表示する。
SHOW CONFIGURATION	システム上の LAN デバイスの一覧を表示する。
SHOW DEVICE	LAN 運用時デバイス・データベースの情報を表示する。
SHOW DLL または MOPDLL	MOP ダウンライン・ロード・サービスの現在の状態を表示する。
SHOW LOG	最近のダウンライン・ロード・アクティビティを表示する。
SHOW NODE	LAN 運用時ノード・データベースの情報を表示する。
SPAWN	現在のプロセスのサブプロセスを作成する。
TRIGGER NODE	遠隔ノードに対してリポート要求を出す。
UPDATE DEVICE	主にデバイスをリセットするために、デバイスをアップデートする。

LANCP コマンドと修飾子についての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』を参照してください。

10.4.3 LANCP のその他の機能

SPAWN コマンドを使用すると、現在のプロセスのサブプロセスを作成できます。SPAWN コマンドは、サブプロセスのコンテキストを現在のプロセスからコピーします。これにより、LANCP を一時的に終了しても、再開時に LANCP を再起動する必要はありません。

SPAWN コマンドの形式は次のとおりです。

SPAWN [コマンド文字列]

LANCP ユーティリティを設定して、LANCP 内でコマンド・ファイルからコマンドを実行できます。LANCP ユーティリティは、コマンド・ファイルを頭に @ の付くファイル名として認識します。省略時のファイル名の拡張子は .COM です。

10.5 LAN デバイスの管理

LAN デバイスの管理には、デバイスの特性の表示と、デバイス・パラメータの設定が含まれます。LANCP ユーティリティを使用して、表 10-7 「LAN デバイス」に示す LAN デバイスのタイプに対してパラメータを設定することができます。

表 10-7 LAN デバイス

LAN	説明
イーサネット	媒体のタイプ (接続ケーブルのタイプ) および接続速度 (イーサネット, FastEthernet, またはギガビット・イーサネット) の選択を可能にする。 全二重動作 (同等のデバイス間またはデバイスとスイッチ間のポイント・ツー・ポイント動作) を可能にする。
FDDI	全二重動作を可能にする。FDDI リング・パラメータの選択を可能にする。

表 10-7 LAN デバイス (続き)

LAN	説明
トークン・リング	トークン・リング・パラメータの設定、およびソース・ルーティングと機能アドレス・マッピングの定義を可能にする。
全タイプ	受信バッファ数などの汎用パラメータの設定を可能にする。
ATM	Emulated LAN (ELAN) パラメータの設定を可能にする。

10.5.1 システム・デバイスの表示

システム上の LAN デバイスを表示するには、次の形式で SHOW CONFIGURATION コマンドを入力します。

SHOW CONFIGURATION

例

LANCP>SHOW CONFIGURATION/USERS

```
LAN Configuration:
Device Parent Medium/User Version Speed Duplex Size MAC Address Current Address Type
EWA0 CSMA/CD X-37 1000 Full 1500 00-30-6E-F4-D0-E1 00-30-6E-F4-D0-E1 BCM5701
EIC0 CSMA/CD X-43 100 Full 1500 00-50-8B-69-A5-E2 AA-00-04-00-CF-4C 82559
EIC2 60-03 DNA Routing 1498
EWB0 CSMA/CD X-37 100 Full 1500 00-30-6E-F2-18-2D 00-30-6E-F2-18-2D BCM5703
EWC0 CSMA/CD X-7 10000 Full 9000 00-0C-FC-00-02-7C 00-0C-FC-00-02-7C XFRAME
EWD0 CSMA/CD X-37 1000 Full 1500 00-30-6E-4A-C5-B9 00-30-6E-4A-C5-B9 BCM5701
```

この例は、8 個の LAN アダプタがあり、そのうちの 1 個が実際に DECnet で使用されているノード上で SHOW CONFIGURATION コマンドを入力したときの出力を示します。

10.5.2 デバイス特性の表示

(運用時デバイス・データベース内の) LAN デバイス についての情報を表示するには、次の形式で SHOW DEVICE コマンドを入力します。

SHOW DEVICE デバイス名 [/修飾子,...]

表 10-8 「SHOW DEVICE コマンド修飾子」に、SHOW DEVICE コマンドの修飾子を簡単に説明します。



注意:

修飾子を指定しなければ、ユーティリティは情報を追加せずに一致するデバイスを表示します。

表 10-8 SHOW DEVICE コマンド修飾子

修飾子	説明
/ALL	デバイス名が一致するすべてのデバイスを表示する。
/CHARACTERISTICS または /PARAMETERS	デバイスについて状態と関連情報を表示する。
/COUNTERS	デバイス・カウンタを表示する。
/DLL または /MOPDLL	ダウンライン・ロード特性を表示する。
/INTERNAL_COUNTERS	内部カウンタを表示する。省略時の設定では、ゼロ・カウンタは表示しない。ゼロ・カウンタを含むすべてのカウンタを表示するには、追加の修飾子 /ZERO を使用する。デバッグ・カウンタを表示するには、追加の修飾子 /DEBUG を使用する。
/MAP	トークン・リングの機能アドレス・マッピング・テーブルの現在の構成を表示する。

表 10-8 SHOW DEVICE コマンド修飾子 (続き)

修飾子	説明
/MESSAGES	LAN ドライバからのコンソール・メッセージを、LAN ドライバ内部カウンタの一部として表示する。
/OUTPUT=file-name	指定されたファイル名に出力する。
/REVISION	LAN ドライバとデバイスの現在のリビジョン情報を表示する (可能な場合)。
/SR_ENTRY	トークン・リングの現在のソース・ルーティング・キャッシュ・テーブルの内容を表示する。
/TRACE	LAN ドライバのトレース・データを表示する。

例

1. LANCP> SHOW DEVICE/COUNTERS EXA0

```
Device Counters EXA0:
      Value Counter
      -----
      259225 Seconds since last zeroed
      5890496 Data blocks received
      4801439 Multicast blocks received
      131074 Receive failure
      764348985 Bytes received
      543019961 Multicast bytes received
      3 Data overrun
      1533610 Data blocks sent
      115568 Multicast packets transmitted
      122578 Blocks sent, multiple collisions
      86000 Blocks sent, single collision
      189039 Blocks sent, initially deferred
      198120720 Bytes sent
      13232578 Multicast bytes transmitted
      7274529 Send failure
      0 Collision detect check failure
      0 Unrecognized frame destination
      0 System buffer unavailable
      0 User buffer unavailable
```

この SHOW DEVICE コマンドは、イーサネット・デバイス EXA0 のカウンタを表示する。

2. LANCP> SHOW DEVICE/MAP ICA0

```
Multicast to Functional Address Mapping ICA0:
Multicast address Functional Address Bit-Reversed
-----
09-00-2B-00-00-04 03-00-00-00-02-00 C0:00:00:00:40:00
09-00-2B-00-00-05 03-00-00-00-01-00 C0:00:00:00:80:00
CF-00-00-00-00-00 03-00-00-08-00-00 C0:00:00:10:00:00
AB-00-00-01-00-00 03-00-02-00-00-00 C0:00:40:00:00:00
AB-00-00-02-00-00 03-00-04-00-00-00 C0:00:20:00:00:00
AB-00-00-03-00-00 03-00-08-00-00-00 C0:00:10:00:00:00
09-00-2B-02-00-00 03-00-08-00-00-00 C0:00:10:00:00:00
09-00-2B-02-01-0A 03-00-08-00-00-00 C0:00:10:00:00:00
AB-00-00-04-00-00 03-00-10-00-00-00 C0:00:08:00:00:00
09-00-2B-02-01-0B 03-00-10-00-00-00 C0:00:08:00:00:00
09-00-2B-00-00-07 03-00-20-00-00-00 C0:00:04:00:00:00
09-00-2B-00-00-0F 03-00-40-00-00-00 C0:00:02:00:00:00
09-00-2B-02-01-04 03-00-80-00-00-00 C0:00:01:00:00:00
09-00-2B-02-01-07 03-00-00-02-00-00 C0:00:00:40:00:00
09-00-2B-04-00-00 03-00-00-04-00-00 C0:00:00:20:00:00
09-00-2B-02-01-00 03-00-00-00-08-00 C0:00:00:00:10:00
09-00-2B-02-01-01 03-00-00-00-10-00 C0:00:00:00:08:00
09-00-2B-02-01-02 03-00-00-00-20-00 C0:00:00:00:04:00
```

```

03-00-00-00-00-01  03-00-00-00-00-01  C0:00:00:00:00:80
03-00-02-00-00-00  03-00-02-00-00-00  C0:00:40:00:00:00

```

この SHOW DEVICE コマンドは、トークン・リング・デバイス ICA0 のマッピング情報を表示する。

3. LANCP> SHOW DEVICE/PARAM IRA0

```

Device Parameters IRA0:
      Value  Parameter
      -----
      Normal  Controller mode
      External Internal loopback mode
00-00-93-58-5D-32 Hardware LAN address
      Token Ring Communication medium
      Enabled  Functional address mode
      No       Full duplex enable
      No       Full duplex operational
      16      Line speed (megabits/second)
      16 Mbps Ring speed
      STP     Line media
      Enabled  Early token release
      Disabled Monitor contender
      200     SR cache entries
      2       SR discovery timer
      60     SR Aging Timer
      Enabled  Source routing
      3       Authorized access priority
AA-00-04-00-92-FF Upstream neighbor
      0       Ring number

```

この SHOW DEVICE コマンドは、トークン・リング・デバイス IRA0 の状態とパラメータ情報を表示する。

4. LANCP> SHOW DEVICE/REVISION EWF0

```

Device Revisions EWF0:
      Value          Component
      02000041      Device hardware revision
      08020110 00000004 Port driver revision
      08020172 00000001 LAN common routines revision

```

この SHOW DEVICE コマンドは、イーサネット・デバイス EWF0 のリビジョン情報を表示する。

5. LANCP> SHOW DEVICE/SR_ENTRY ICA0

```

Source Routing Cache Table ICA0:
      LAN address      State      XmtTmo      RcvTmo      StaleTmo      DiscvTmo
      -----
AA-00-04-00-92-FF  LOCAL      00000028  00000028  00000245  00000000

```

この SHOW DEVICE コマンドは、トークン・リング・デバイス ICA0 のソース・ルーティング・エントリ情報を表示する。

10.5.3 デバイス特性の設定

LAN デバイスはすべて、パラメータの集まりで特徴付けられます。パラメータは、デバイスが接続されている媒体の LAN デバイスの操作特性を定義します。

LAN デバイスのパラメータを直接設定するには、LANCP> プロンプトで SET DEVICE コマンドを入力します。LANCP ユーティリティは、LANACP サーバ・プロセスにこのコマンドを発行し、LANACP サーバ・プロセスは適切な QIO を発行してデバイス特性を設定します。

SET DEVICE コマンドの形式は次のとおりです。

SET DEVICE デバイス名 [/修飾子]

以降の項では、一般的な LAN デバイスや、特定の LAN デバイスに直接適用される SET DEVICE コマンドの修飾子について説明します。コマンドの修飾子は、以下のカテゴリに分類されます。

- 汎用
- イーサネット・デバイス
- LAN フェイルオーバ・デバイス
- FDDI デバイス
- トークン・リング・デバイス
- ATM デバイス

10.5.3.1 イーサネット・デバイスでの、DEFINE DEVICE コマンドおよび SET DEVICE コマンドの修飾子

この項で説明する修飾子は、イーサネット・デバイスで使用できます。

- /[NO]AUTONEGOTIATE
リンク設定を確認するための自動ネゴシエーションを有効または無効にします。自動ネゴシエーションをサポートしていないスイッチまたはデバイスに接続する場合には、リンクの自動ネゴシエーションを無効にしておく必要があります。
- /[NO]FLOW_CONTROL
LAN デバイスのフロー制御を有効または無効にします。
- /[NO]FULL_DUPLEX
LAN デバイスの全二重動作を有効または無効にします。この修飾子を使用して全二重動作をさせるためには、デバイスまたはネットワーク・ハードウェアの追加の設定が必要になる場合があります。全二重動作を有効にできるデバイスもありますが、設定を変更できないデバイスもあります。/NOFULL_DUPLEX 修飾子は、全二重動作を無効にします。
- /[NO]JUMBO
LAN デバイスでのジャンボ・フレームの使用を有効にします。ギガビット・イーサネット NID だけが、ジャンボ・フレームをサポートしています。
- /MEDIA= 値
ケーブル接続を選択します。通常この選択は、デバイスの初期化の際に、ツイスト・ペアを選択する制限付きの自動検出アルゴリズムを使用して行われます。しかし、ツイスト・ペアが機能していないと見なされた場合は、AUI (Attachment Unit Interface) にフェイルオーバします。この後、ケーブルを変更するためには、システムをリブートする必要があります。このコマンドを使用すると、リブートせずにケーブルの選択を変更することができます。
使用できる値は、AUI (10Base2, 10Base5), TWISTEDPAIR (10BaseT), AUTOSENSE (制限付きの自動検出アルゴリズムを再実行する) です。省略時の値は、AUTOSENSE です。
DE435 などの一部のデバイスでは、10Base2 と 10Base5 (シンワイヤとシックワイヤ) の間で切り替えるために、イーサネット・カードのジャンパを変更する必要があります。
DE434, DE436, および DE500 などのデバイスでは、ツイスト・ペア接続しか備えていません。
- /SPEED= 値
LAN の速度を設定します。使用できる値は、10, 100, 1000, または自動ネゴシエーション (イーサネットの場合は 10M ビット/秒, FastEthernet の場合は 100M ビット/秒, ギガビット・イーサネットの場合は 1000M ビット/秒を選択する) です。自動ネゴシエーションを選択すると、LAN ドライバは自動ネゴシエーションを再実行します。

10.5.3.2 LAN フェイルオーバー・デバイスでの、DEFINE DEVICE コマンドおよび SET DEVICE コマンドの修飾子

ここで説明する修飾子は、LAN フェイルオーバー・デバイスで使用できます。/PRIORITY を除き、修飾子はフェイルオーバー・セット内の LAN デバイスではなく、論理 LAN デバイス (LL) に適用されます。

- /DISABLE
LAN のフェイルオーバー・セットを使用不能にします。使用不能にすると、LAN デバイスをフェイルオーバー・セットに追加したり、フェイルオーバー・セットから削除したりできません。
- /ENABLE
フェイルオーバー・セットを使用可能にし、論理 LAN デバイスを起動します。LAN フェイルオーバー・ドライバは、LAN フェイルオーバー・セットから LAN デバイスをアクティブなデバイスとして選択してから、論理 LAN デバイスへの I/O を可能とします。
- [NO]/FAILOVER_SET=(デバイス名 [,...])
追加または削除する LAN フェイルオーバー・セットのメンバを指定します。
- /PRIORITY= 値
LAN デバイスのフェイルオーバー優先順位を設定します。アクティブにする LAN デバイスを選択する際には、優先順位が最高の LAN フェイルオーバー・デバイスが優先的に選択されます。
- /SIZE= 値
LAN フェイルオーバー・デバイスのパケット・サイズを設定します。使用できる値は、STANDARD (省略時の値) または JUMBO です。
 - STANDARD は、最大のイーサネット・パケット・サイズである、1518 バイトです。
 - JUMBO は、ギガビット・イーサネット・デバイスで利用できる、大きなパケット・サイズです。JUMBO が利用できるのは、LAN フェイルオーバー・セット内のすべての LAN デバイスがギガビット・デバイスである場合だけです。LAN フェイルオーバー・デバイスに指定されたサイズは、LAN フェイルオーバー・セットのメンバに設定されたサイズより優先されます。つまり、LAN フェイルオーバー・デバイスによりサイズが決まった場合は、LAN デバイスの JUMBO フレーム設定は意味を持たなくなります。
- /SWITCH (SET DEVICE のみ)
LAN フェイルオーバー・セットの他のメンバへ、強制的に LAN をフェイルオーバーさせます。このコマンドを使用して、あるデバイスから他のデバイスへ切り替えることで、LAN のフェイルオーバー動作をテストできます。

10.5.3.3 FDDI デバイスでの、DEFINE DEVICE コマンドおよび SET DEVICE コマンドの修飾子

ここで説明する修飾子は、FDDI デバイスで使用できます。

- /RING_PURGER
FDDI デバイスのリングのページ処理を有効にします。
- /TOKEN_ROTATION
FDDI リングに対して要求されたトークン・ローテーション時間を設定します。
- /TOKEN_TIMEOUT
FDDI リングに対して制限付きのトークン・タイムアウト時間を設定します。
- /TRANSMIT_TIMEOUT
FDDI デバイスに対して有効な転送時間を設定します。

10.5.3.4 トークン・リング・デバイスでの、DEFINE DEVICE コマンドおよび SET DEVICE コマンドの修飾子

ここで説明する修飾子は、トークン・リング・デバイスで使用できます。

- /AGING_TIMER= 値
トークン・リングのソース・ルーティング・キャッシュ・エントリが無効とマークされるまでの時間を秒単位で設定します。このタイマは、遠隔ノードとの送信または受信トラフィックがこの時間内になかったときに満了します。省略時の値は、60 秒です。
アイドル状態で、古い状態と既知の状態の間を行き来する場合は、この値を大きくしてください。この値を小さくしすぎると、不要なエクスプローラ・トラフィックが LAN 上を流れる可能性があります。
- /CACHE_ENTRIES= 値
トークン・リングのソース・ルーティング・アドレス・エントリのキャッシュ用に確保するエントリの数を設定します。省略時の値は、200 エントリです。多数のシステムと直接通信しているシステムでは、この値を大きくしてください。
- /CONTENDER
/NOCONTENDER (省略時の設定)
トークン・リング・デバイスがリングに加わるときに、モニタ・コンテンション・プロセスに加わることを指定します。/NOCONTENDER 修飾子は、このデバイスが現在のリング・サーバと競合しないように指定します。
- /DISCOVERY_TIMER=value
ソース・ルーティング・ルート検出プロセスを実行しているときに、遠隔ノードからの応答を待つ時間を秒数で設定します。省略時の値は 2 秒です。
大きな LAN で、応答が遅いノードが存在する場合は、LAN 上を流れるエクスプローラ・トラフィックの量を減らすために、この値を大きくしなければならないことがあります。
- /EARLY (省略時の設定)
/NOEARLY
デバイス上で Early Token Release を有効にします。/NOEARLY 修飾子は、Early Token Release を無効にします。
- /MAP=(MULTICAST_ADDRESS= アドレス,FUNCTIONAL_ADDRESS= アドレス)
/NOMAP=(MULTICAST_ADDRESS= アドレス)
標準のマルチキャスト・アドレスを、機能アドレスにマッピングします。トークン・リング・デバイスは、IEEE 802 標準の、グローバル定義グループ・アドレスはサポートしません。トークン・リング・デバイスは、機能アドレスをサポートしません。機能アドレスは、31 通りの値を使用できる、ローカル管理のグループ・アドレスです。各機能アドレスは、アドレスの 3 バイト目から 6 バイト目までで 1 ビットを設定し、バイト 1 とバイト 2 は 03-00 です (ビット逆順形式では C0:00)。
/NOMAP=(MULTICAST_ADDRESS= アドレス) 修飾子は、指定されたアドレスに設定されていたマッピングをクリアします。機能アドレスは、次のように指定します。
 - MULTICAST_ADDRESS 引数には、標準の 6 バイト・マルチキャスト・アドレスを指定します。
 - FUNCTIONAL_ADDRESS 引数には、機能アドレスの最後の 4 バイトだけを指定します (先行する 03-00 のバイトは、自動的に付加されます)。
 - ハイフンで区切られた 16 進バイト文字として指定されたアドレス変数は、アドレスの標準形式です。ビット逆順形式のアドレスを指定するには、区切り文字としてコロンを使用してください。たとえば、マルチキャスト・アドレス CB-00-01-02-03-04 を、トークン・リング・デバイス IRA0 の機能アドレス 03-00-00-80-00-00 にマッピングするには、次のコマンドを入力します。

§ SET DEVICE IRA0/MAP=(MULTI=CB-00-01-02-03-04,FUNCT=00:01:00:00)

省略時のアドレス・マッピングについては、次の表を参照するか、コマンド SHOW DEVICE/MAP デバイス名を入力してください。

表 10-9 トークン・リング・デバイスの、省略時の機能アドレス

マルチキャスト・アドレス	機能アドレス	説明
09-00-2B-02-01-0B	03-00-10-00-00-00	DNA フェーズ IV プライム未知デスクティネーション
09-00-2B-00-00-07	03-00-20-00-00-00	PCSA NETBIOS エミュレーション
09-00-2B-00-00-0F	03-00-40-00-00-00	LAT サービス通知
09-00-2B-02-01-04	03-00-80-00-00-00	LAT サービス送信請求
09-00-2B-02-01-07	03-00-00-02-00-00	LAT Xwindow サービス送信請求
09-00-2B-04-00-00	03-00-00-04-00-00	LAST
09-00-2B-02-01-00	03-00-00-00-08-00	DNA ネーム・サービス通知
09-00-2B-02-01-01	03-00-00-00-10-00	DNA ネーム・サービス送信請求
09-00-2B-02-01-02	03-00-00-00-20-00	DNA タイム・サービス
03-00-00-00-00-01	03-00-00-00-00-01	NETBUI エミュレーション
03-00-02-00-00-00	03-00-02-00-00-00	RIPL

- /MEDIA= 値
ケーブル・タイプを自動的に検出しないデバイスについて、トークン・リング MAU (Media Access Unit) に、アダプタを接続するために使用されているケーブルのタイプを選択します。使用できる値は、シールドなしツイスト・ペア (UTP)、またはシールド付きツイスト・ペア (STP) です。省略時の値は、STP です。
- /SOURCE_ROUTING (省略時の設定)
/NOSOURCE_ROUTING
トークン・リング・デバイス上でのソース・ルーティングを有効にします。LAN 内にリングが 1 つしかない場合や、透過的なブリッジを使用している場合は、/NOSOURCE_ROUTING 修飾子を使用してソース・ルーティングをオフにしてください。
- /SPEED= 値
トークン・リング LAN の速度を設定します。使用できる値は 4 と 16 で、それぞれ 4M ビット/秒と 16M ビット/秒を表します。トークン・リングでの省略時の値は、LAN アダプタがこのパラメータを設定するための非揮発性の機構をサポートしている場合を除いて、16 です。
- /SR_ENTRY=(LAN_ADDRESS= アドレス, RI= ルーティング情報)
/NOSR_ENTRY=(LAN_ADDRESS= アドレス)
特定のノードに対して、特定のソース経由のルートを経路を静的に定義します。省略時の設定では、ルートは指定されていません。このキャッシュは、使用中は有効になっており、エージング・タイマが満了するまで有効のままです。
この修飾子は、大きな LAN トポロジーで通信上の障害を隔離するための最後の手段としてだけ使用してください。
/NOSR_ENTRY=(LAN_ADDRESS= アドレス) 修飾子は、以前に定義された、静的ソース・ルーティングの経路をクリアします。アドレスは、アドレスの標準形式で指定する 6 バイトの LAN アドレスです (ハイフンで区切られた 16 進のバイト文字として指定)。区切り文字としてコロンを使用すると、ビット逆順形式のアドレスであることを示します。

ルーティング情報は、一連の 2 バイトの 16 進文字 (各バイトはハイフンで区切る) として指定されるソース・ルーティング・フィールドです。このフィールドは、2 バイトのルーティング制御フィールドの後に、ホップで使用されるリング番号とブリッジ番号をそれぞれ含んでいる 14 個までの 2 バイトのセグメント識別子を続けたものです。

10.5.3.5 修飾子 (ATM デバイス)

ここで説明する修飾子は、ATM (Asynchronous Transfer Mode) デバイスで使用できます。

- /ATMADDRESS=LES

ATM のための LAN エミュレーション・サーバ (LES) アドレスを設定します。通常このアドレスはユーザが指定するものではないため、特定のアドレスを指定したい場合にのみこの修飾子を使用します。省略時の設定では、アドレスは LES の構成サーバでソフトウェアにより決定されます。

/ATMADDRESS=LES 修飾子の構文は次のとおりです。

```
SET DEVICE/ATMADDRESS= ([NO] LES=ATM サーバ)
```

- /ATMADDRESS=ARP

ATM 上の Classical IP のためのアドレス解決プロトコル (ARP) サーバのアドレスを設定します。ローカル・ホストが ARP サーバでない場合は、LIS を有効にするまではこの修飾子を指定する必要があります。

/ATMADDRESS =ARP 修飾子の構文は次のとおりです。

```
SET DEVICE/ATMADDRESS= (ARP=atm_arp_server)
```

- /CLIP

ATM (RFC1577) 上の Classical Internet Protocol (CLIP) を設定します。CLIP 修飾子は、データ・リンク・レベルのデバイスを、論理 IP サブネット (LIS) のクライアントまたはサーバあるいはその両方としてインプリメントします。これにより、IP プロトコルが、ATM ネットワーク上でイーサネット・フレームを送信できるようになります。/CLIP=ENABLE コマンドを使用すると、システムが LIS に参加できるようになります。/CLIP=DISABLE コマンドを使用すると、クライアントは論理 IP サブネットから離れます。

LIS ではサーバが必要になりますが、サーバは 1 つのサブネットにつき 1 つだけです。サブネット間の通信は、ルータ経由でのみ可能になります。各 ATM アダプタには、クライアントが 1 つだけ存在します。

標準のインターネット・ドット表記法を使用した /CLIP 修飾子の構文は、次のとおりです。

```
SET DEVICE/CLIP=(ip_subnet-a.b.c.d,  
ip_address=a.b.c.c, parent=devnam, name="ipsubnet name",  
enable,disable, type=client | server)
```

/CLIP の構文の意味を、表 10-10 「/CLIP の構文」に示します。

表 10-10 /CLIP の構文

オプション	意味
ip_address	CLIP クライアントの IP アドレスを指定する。
ip_subnet	CLIP クライアントのサブネット・マスクを指定する。
parent=devnam	親デバイス名を指定する。
name	操作および診断メッセージで使用するための LIS の名前を指定する。
type=client	classical IP クライアントのみを起動する。これが省略時の設定になる。
type=server	classical IP サーバを起動する。各 LIS で使用できるサーバは 1 つだけで、サーバを先に起動する必要がある。
type=(server,client)	classical IP サーバとクライアントを起動する。

/CLIP のキーワードとその意味を、表 10-11 「/CLIP のキーワード」 に示します。

表 10-11 /CLIP のキーワード

キーワード	意味
Enable	論理 IP サブネットに参加する。
Disable	クライアントが論理 IP サブネットから離れるようにする。

- /ELAN

/ELAN 修飾子は、ENABLE と DISABLE のいずれかの値をとります。キーワード STARTUP とともに /ELAN=ENABLE を指定すると、LANACP が開始するときに LAN エミュレーションがロードされます。/ELAN=DISABLE では、ENABLE の場合と同じパラメータを使用できます。

/ELAN 修飾子の構文は次のとおりです。

```
DEFINE DEVICE/ELAN=(parent=parent device, name="ELAN NAME to join",  
size=1516 type=CSMACD Enable,Disable, description="description  
string,")
```

/ELAN の構文の意味は、表 10-12 「/ELAN の構文」 のとおりです。

表 10-12 /ELAN の構文

オプション	意味
parent	ATM アダプタ・デバイスの名前。たとえば、DAPCA の親デバイスは HWn0 (n はコントローラの番号) で、DGLTA の親デバイスは HCn0 (n はコントローラの番号)。
name	特定の ELAN に参加したい場合、オプションとして指定することができる。省略時の設定は、null。
size	参加したい LAN の最大フレーム・サイズ。有効なサイズは 1516 バイト、4544 バイト、9234 バイトのいずれか。省略時の値は 1516。
type	現在 CSMACD のみがサポートされており、これが省略時の値。
description	ELAN に注釈をつけるために使用する。表示のためだけに使用される。

/ELAN のキーワードとその意味を、表 10-13 「/ELAN のキーワード」 に示します。

表 10-13 /ELAN のキーワード

キーワード	意味
Enable	指定したエミュレートされた LAN で参加を開始する。ドライバがロードされていない場合はロードする。
Disable	クライアントに、エミュレートされた LAN から離れさせる。

- /PVC=(vci[,...])
/[NO]PVC=(vci[,...])

ATM 上の Classical IP クライアントが使用するパーマネント仮想サーキット (PVC) を設定する。この修飾子は、オプション。

PVC のリストは、CLIP クライアントで使用するために定義される。このコマンドを使用した後で CLIP クライアントを有効にすること。ATM スイッチでは、PVC は手動で設定する必要がある。

vci は、使用する PVC の VCI (仮想サーキット ID) です。

例

1. LANCP> SET DEVICE/CONTENDER/MEDIA=UTP/NOEARLY/SOURCE ICA0

このコマンドで、モニタのコンテンション、UTP ケーブル・メディア、ソース・ルーティングが使用可能になり、トークン・リング・デバイス ICA0 の Early Token Release が使用不能になる。

2. LANCP> SET DEVICE/MEDIA=TWIST EWBO

このコマンドは、媒体のタイプを 2 番目の Tulip イーサネット・デバイスのツイストペアに設定する。

3. LANCP> SET DEVICE/ALL/MIN_BUFFERS=12

このコマンドは、全 LAN デバイスの受信バッファ数を 12 以上に設定する。

4. LANCP> DEFINE DEVICE EXA0/MOPDLL=(ENABLE,EXCLUSIVE)

このコマンドは LAN デバイス EXA0 を定義して、排他的モードで LANACP MOP ダウンライン・ロード・サービスを許可する。KNOWNCLIENTSONLY および SIZE 特性の設定値は変更されない。デバイス・エントリが LAN パーマネント・デバイス・データベースに現在存在しない場合には、この設定値が省略時の値に設定される。

5. LANCP> DEFINE DEVICE/ALL/MOPDLL=NOEXCLUSIVE

このコマンドは、LAN パーマネント・デバイス・データベースに定義されている全デバイスに対して、非排他的モードの LANACP MOP ダウンライン・ロード・サービスを設定する。

6. LANCP> SET DEVICE EXA0/MOPDLL=(ENABLE,NOEXCLUSIVE)

LANCP> SET DEVICE FXA0/MOPDLL=(ENABLE,EXCL,KNOWN)

このコマンドは次の状態の LANACP MOP ダウンライン・ロード・サービスを許可する。

- 非排他的モードの LAN デバイス EXA0
- 既知のクライアントのみに対し、排他的モードの LAN デバイス FXA0

10.6 LAN デバイス・データベースの管理

LAN の運用時およびパーマネント・デバイス・データベースには、システムに存在する LAN デバイスごとに 1 つのエントリがあります。LAN 運用時デバイス・データベースの各エントリは、デバイス情報および MOP ダウンライン・ロード・カウンタ情報を含みます。LAN パーマネント・デバイス・データベースの各エントリに含まれるデバイス情報は、LANACP LAN サーバ・プロセスの起動時に運用時データベースを作成するのに使用されます。

通常、各データベースは同じデバイスを含んでいます。ただし、パーマネント・データベースには、システムにまだ構成されていない、またはインストールされていないデバイスのエントリが含まれる場合があります。LANACP LAN サーバ・プロセスは、運用時デバイス・データベースを管理します。LANCP ユーティリティは、パーマネント・デバイス・データベースを管理します。どちらのデータベースも LANCP ユーティリティ・コマンドで操作できますが、次に示すように、操作できる内容はユーザ特権によって異なります。

- 特権ユーザは、各データベースに対するデバイス・エントリの追加と削除、MOP ダウンライン・ロード・サービスの許可と禁止、LAN デバイスについての MOP ダウンライン・ロード・カウンタ情報のクリアを指定できる。
- 非特権ユーザは、MOP ダウンライン・ロード状態およびカウンタ情報を表示できる。

以降の各項では、LAN パーマネント・デバイス・データベースおよび運用時デバイス・データベースへのデバイスの入力と削除の方法、および MOP ダウンライン・ロード・サービスの許可と禁止の設定方法について説明します。

10.6.1 LAN デバイス・データベース内のデバイスの表示

LAN パーマネント・デバイス・データベースの情報を表示するには、LIST DEVICE コマンドを次の形式で入力します。

LIST DEVICE **デバイス名** [/修飾子,...]

LAN 運用時デバイス・データベースの情報を表示するには、SHOW DEVICE コマンドを次の形式で入力します。

SHOW DEVICE **デバイス名** [/修飾子,...]

表 10-14 「LIST DEVICE および SHOW DEVICE コマンド修飾子」に、LIST DEVICE 修飾子と SHOW DEVICE 修飾子について簡単に説明します。

表 10-14 LIST DEVICE および SHOW DEVICE コマンド修飾子

修飾子	説明
/ALL	デバイス名が一致するすべてのデバイスをリスト、または表示します。
/CHARACTERISTICS または /PARAMETERS	デバイスについての状態および関連情報を表示する。
/COUNTERS ¹	デバイス・カウンタを表示する。
/DLL または /MOPDLL	ダウンライン・ロード特性をリスト、または表示します。
/MAP	トークン・リングの機能アドレス・マッピング・テーブルの現在の構成を表示する。
/OUTPUT= ファイル名	指定されたファイルを作成し、そのファイルに出力する。
/REVISION ¹	アダプタの現在のファームウェア・リビジョンを表示する (可能な場合)。
/SR_ENTRY	現在のトークン・リングのソース・ルーティング・キャッシュ・テーブルの内容を表示する。

¹ SHOW DEVICE のみ



注意:

修飾子を指定しなければ、一致するデバイスが追加情報を伴わずに表示されます。

10.6.2 LAN デバイス・データベースへのデバイスの入力

LAN パーマネント・デバイス・データベースにデバイスを入力したり、既存のエントリを変更するには、次の形式で DEFINE DEVICE コマンドを入力します。

DEFINE DEVICE **デバイス名** [/修飾子,...]

LAN 運用時デバイス・データベースにデバイスを入力したり、既存のエントリを変更するには、次の形式で SET DEVICE コマンドを入力します。

SET DEVICE **デバイス名** [/修飾子,...]

10.6.3 LAN デバイス・データベースからのデバイス・データの削除

LAN パーマネント・デバイス・データベースからデバイス・データを削除するには、次の形式で PURGE DEVICE コマンドを入力します。

PURGE DEVICE **デバイス名** [/修飾子]

削除するデータのタイプを選択する修飾子がない場合、デバイス・エントリ全体が削除されません。

表 10-15 「PURGE DEVICE 修飾子」で、PURGE DEVICE 修飾子について簡単に説明します。

表 10-15 PURGE DEVICE 修飾子

修飾子	説明
/ALL	LAN パーマネント・デバイス・データベースからすべての LAN デバイスのデータを削除する。デバイス名を指定すると、一致する LAN デバイスがすべて選択される。たとえば、E を指定するとすべてのイーサネット・デバイス、F を指定すると FDDI デバイス、I を指定するとトークン・リング・デバイス、EW を指定するとすべてのイーサネット PCI Tulip デバイスが選択される。
/CHARACTERISTICS または /PARAMETERS	速度、二重モード、およびその他のデバイス・パラメータなど、LAN デバイスのデバイス特性設定を削除する。
/DLL または /MOPDLL	LAN デバイス用の MOP ダウンライン・ロード設定を削除する。
/TRACE	LAN デバイスのドライバ・トレース設定を削除する。

例

1.

LANCP> **PURGE DEVICE/ALL**

このコマンドは、LAN パーマネント・デバイス・データベースから全デバイスのデータを削除します。

10.7 LAN ノード・データベースの管理

LAN 運用時ノード・データベースおよびパーマネント・ノード・データベースには、定義された各 LAN ノードに対して 1 つのエントリがあります。LAN 運用時ノード・データベースの各エントリは、ノード情報と MOP ダウンライン・ロード・カウンタ情報を含みます。LAN パーマネント・ノード・データベースの各エントリに含まれるノード情報は、LANACP LAN サーバ・プロセスの開始時に、運用時データベースを作成するのに使用されます。

通常、各データベースは同じノードを含んでいます。LANACP LAN サーバ・プロセスは運用時ノード・データベースを管理します。LANCP ユーティリティはパーマネント・ノード・データベースを管理します。どちらのデータベースも LANCP ユーティリティ・コマンドで操作できます。ただし次に示すように、操作できる内容はユーザ特権によって異なります。

- 特権ユーザは、各データベースに対するノード・エントリの追加と削除、LAN ノードについての MOP ダウンライン・ロード・カウンタ情報のクリアを指定できる。
- 非特権ユーザは、ノード情報に加えて、MOP ダウンライン・ロード状態とカウンタ情報を表示できる。

以降の各項では、LAN パーマネント・ノード・データベースおよび運用時ノード・データベースへのノードの入力と削除の方法について説明します。

10.7.1 LAN ノード・データベース内のノードの表示

LAN パーマネント・ノード・データベースの情報を表示するには、次の形式で LIST NODE コマンドを入力します。

LIST NODE ノード名 [/ALL]

LAN 運用時ノード・データベースの情報を表示するには、次の形式で SHOW NODE コマンドを入力します。

SHOW NODE ノード名 [/ALL]

LIST NODE コマンドと SHOW NODE コマンドの場合、/ALL 修飾子を指定すると、LAN パーマネント・ノード・データベースまたは運用時ノード・データベースの全ノードの情報が表示されます。

10.7.2 LAN ノード・データベースへのノードの入力

LAN パーマネント・ノード・データベースにノードを入力したり、既存のエントリを変更するには、次の形式で DEFINE NODE コマンドを入力します。

DEFINE NODE ノード名 [/修飾子,...]

LAN 運用時ノード・データベースにノードを入力したり、既存のエントリを変更するには、次の形式で SET NODE コマンドを入力します。

SET NODE ノード名 [/修飾子,...]

表 10-16 「DEFINE NODE および SET NODE コマンド修飾子」で、DEFINE NODE および SET NODE コマンド修飾子について簡単に説明します。

表 10-16 DEFINE NODE および SET NODE コマンド修飾子

修飾子	説明
/ADDRESS= ノード・アドレス	LAN アドレスをノード名に対応付ける。
/ALL	LAN パーマネント・ノード・データベースまたは運用時ノード・データベースの全ノードに対してデータを定義する。
/BOOT_TYPE=VAX_SATELLITE ALPHA_SATELLITE OTHER	ダウンライン・ロード要求に必要な処理のタイプを指定する。
/FILE= ファイル指定	ダウンライン・ロード要求にファイル名が含まれない場合、提供するファイル名を指定する。
/PERMANENT_DATABASE (SET コマンドのみ)	LAN 運用時ノード・データベース内のノード・エントリを、パーマネント・データベースに現在設定されているデータで更新する。
/ROOT= ディレクトリ指定	ファイル名に関連するディレクトリ指定を指定する。
/SIZE= 値	各ダウンライン・ロード・メッセージのファイル・データ部分のサイズをバイト数で指定する。
/V3	このノードからの MOP バージョン 3 のブート要求に限って応答することを、サーバに対して設定する。
/VOLATILE_DATABASE (DEFINE NODE コマンドのみ)	LAN パーマネント・ノード・データベース内のノード・エントリを、運用時データベースに現在設定されているデータで更新する。

例

1. DEFINE NODE GALAXY /ADDRESS=08-00-2B-11-22-33 -
/FILE=NISCS_LOAD.EXE -
/ROOT=\$64\$DIA14:<SYS10.> -
/BOOT_TYPE=VAX_SATELLITE

このコマンドは、LAN パーマネント・ノード・データベース内のノード GALAXY を、OpenVMS Cluster システムの VAX サテライトとしてブートすることを設定する。

NISCS_LOAD.EXE ファイルは実際には \$64\$DIA14:<SYS10.SYSCOMMON.SYSLIB> にある。<SYSCOMMON.SYSLIB> は LANACP LAN サーバ・プロセスによって提供され、ルーート定義には含まれない。

2. DEFINE NODE ZAPNOT /ADDRESS=08-00-2B-11-22-33 -
/FILE=APB.EXE -
/ROOT=\$64\$DIA14:<SYS10.> -
/BOOT_TYPE=ALPHA_SATELLITE

このコマンドは、ノード ZAPNOT を OpenVMS Cluster システムの Alpha サテライトとしてブートすることを設定する。

APB.EXE ファイルは実際には \$64\$DIA14:<SYS10.SYSCOMMON.SYSEXEXE> にある。
<SYSCOMMON.SYSEXEXE> は LANACP LAN サーバ・プロセスによって提供され、ルート定義には含まれない。

**3. SET NODE CALPAL/ADDRESS=08-00-2B-11-22-33 -
/FILE=APB_061.EXE**

このコマンドは、ノード CALPAL を、InfoServer イメージのブート用に設定する。また、ノード CALPAL から受信したロード要求にファイル名がない場合、ロードされるファイルを定義する。

ファイルにディレクトリ指定が含まれないため、論理名 LAN\$DLL でファイルの位置が定義される。ファイル名の使用、または /ROOT 修飾子の使用によって、ディレクトリ指定を指定することができる。

ブート・コマンドの中で明示的にファイル名を指定すると、ノード・データベース・エントリに指定されるファイル名が上書きされることに注意。

10.7.3 LAN ノード・データベースからのノードの削除

LAN パーマネント・ノード・データベースからノードを削除するには、次の形式で PURGE NODE コマンドを入力します。

PURGE NODE ノード名 [/ALL]

LAN 運用時ノード・データベースからノードを削除するには、次の形式で CLEAR NODE コマンドを入力します。

CLEAR NODE ノード名 [/ALL]

PURGE NODE コマンドと CLEAR NODE コマンドの場合、/ALL 修飾子を指定すると、LAN パーマネント・ノード・データベースまたは運用時ノード・データベースの全 LAN ノードを削除します。

10.8 LAN MOP

LANCP および LANACP では、数多くのユーティリティとスタートアップ・コマンド・ファイルを備えており、MOP ダウンライン・ロード・サービスに必要な機能を実現します。これらのユーティリティとファイルは、クラスタ・サテライト、ターミナル・サーバ、コンソール更新イメージやシステム・ソフトウェア更新イメージ (Inforserver ロードの場合) などの、特殊イメージのダウンライン・ロードを求めるシステムをロードします。

10.8.1 DECnet MOP との共存

LAN MOP 環境は、DECnet で提供される機能に類似した機能を実現します。この結果、システム管理者は、DECnet MOP と LAN MOP のいずれかの機能を選択できます。OpenVMS Cluster システムの場合、LAN MOP を選択すると、DECnet を使用せずにクラスタを操作することができます。

LAN MOP は、次の状態で DECnet MOP と共存できます。

- 異なるシステム上で実行している。
たとえば、DECnet MOP サービスが LAN 上のあるシステムで許可されており、LAN MOP は他のシステム上で許可されている状態がこれに相当する。
- 同一システム上の異なる LAN デバイス上で実行している。
たとえば、DECnet MOP サービスが、システム上で使用できる LAN デバイスのサブセット上で許可されており、残りのデバイス上の LAN MOP が許可されている状態がこれに相当する。

- 同一システム上の同一 LAN デバイス上で実行しているが、異なるノード・セットをサービスの対象としている。
たとえば、DECnet MOP と LAN MOP の両方が許可されているが、LAN MOP が対応するノードが限られている状態がこれに相当する。これによって DECnet MOP は、残りのノードに対応できるようになる。

10.8.2 DECnet MOP から LAN MOP への移行

LAN MOP へ移行するには、次の手順を実行します。

1. MOP ダウンライン・ロード・サービスを提供するノードを決定する。これは DECnet のサービスを現在許可しているノードと同じでもかまわない。
2. DCL プロンプトに対して次のコマンドを入力し、LAN パーマネント・デバイス・データベースを作成する。
LANCP> **DEFINE DEVICE/UPDATE**
3. 各クラスタ・サテライト・ノード、および DECnet ノード・データベース内で同様に定義されているその他のノードの、ノード定義を入力することによって、LAN パーマネント・ノード・データベースを作成する。このデータは手操作で入力することも、コマンド・プロシージャ SYS\$EXAMPLES:LAN\$POPULATE.COM を実行し、表示される指示とヘルプに従って入力することもできる。
4. 運用時データベース内で現在許可されている、各 DECnet サーキット上でのサービスを禁止する。
5. LAN パーマネント・デバイス・データベース内の、使用する LAN デバイスごとに、DCL プロンプトに対して次のコマンドを入力することによって、各 LAN デバイス上でのサービスを許可する。

LANCP> **DEFINE DEVICE デバイス名/MOPDLL=ENABLE**

6. 高性能が求められる場合は、1482 バイトのデータ・サイズを選択し、ロード要求がいくつか失敗した場合に限ってこのサイズを小さくする。または、小さなデータ・サイズが必要なクライアントをロードするシステムを 1 つ設定し、他のクライアントをロードするシステムを別に設定する。

DECnet MOP に永久的に戻すには、次の手順に従ってください。

1. 次を入力して、運用時データベース内の MOP サービスを禁止する。

LANCP> **SET DEVICE デバイス名/MOPDLL=DISABLE**

2. 次を入力して、LANCP のパーマネント・データベースにある MOP サービスを禁止する。

LANCP> **DEFINE DEVICE デバイス名/MOPDLL=DISABLE**

3. パーマネント・データベースと運用時データベースにあるそれぞれの DECnet サーキットに対して、サービスを再度許可する。



注意:

LAN MOP でのブート時に追加したノードは、ダウンライン・ロードのターゲットとして DECnet ノード・データベースには入力されず、DECnet MOP に戻った時に更新する必要があります。

10.8.3 CLUSTER_CONFIG_LAN.COM と LAN MOP

サテライトの LAN MOP ブートを行う上で、LANCP を簡単に使用できることを目的として、クラスタ管理コマンド・プロシージャが提供されています。このコマンド・プロシージャが CLUSTER_CONFIG_LAN.COM で、SYS\$MANAGER 内にあり、OpenVMS Cluster システムの構成および再構成を行うためにクラスタ管理者が使用する CLUSTER_CONFIG.COM と、全く同じ目的を持ちます。この 2 つのプロシージャは同じ機能を実行しますが、CLUSTER_CONFIG.COM がダウンライン・ロードに DECnet MOP を使用するのに対して、

CLUSTER_CONFIG_LAN.COM は LAN MOP を使用して、DECnet を使用しません。このため、新たなノードを追加した場合、CLUSTER_CONFIG_LAN.COM がノードの DECnet ノード名とアドレスを求めることはありません。代わりに、SCS ノード名と SCS ノード ID 番号を求めます。

便宜上、CLUSTER_CONFIG.COM をこれまで通り実行することもできます。CLUSTER_CONFIG.COM を実行すると、LANACP の MOP ブートが実行されているかどうかをチェックします。また、DECnet が実行されているかどうかについてもチェックします。LANACP が実行されていて、DECnet が実行されていない場合、CLUSTER_CONFIG.COM は CLUSTER_CONFIG_LAN.COM にディスパッチします。また、CLUSTER_CONFIG.COM が、LANACP と DECnet の両方とも実行されていることを検出すると、LAN MOP が使用されているかどうか、および CLUSTER_CONFIG_LAN.COM をユーザに対して呼び出すかどうかについて、ユーザに問い合わせてきます。

10.8.4 サテライト・ロードのサンプル

MOP ダウンライン・ロード・サービスを許可し、ノード ZAPNOT を定義するための、LANCP ユーティリティに対するコマンドの実行方法を次に示します。

```
set acp/opcom
set device eza0/mopdll=enable
set node ZAPNOT/addr=08-00-2B-33-FB-F2/file=APB.EXE-
        /root=$64$DIA24:<SYS11.>/boot=Alpha
```

次に、LANACP LAN サーバ・プロセスを起動したときに表示される OPCOM メッセージを示します。

```
%%%%%%%%%% OPCOM 10-JAN-2001 06:47:35.18 %%%%%%%%%%%
Message from user SYSTEM on GALAXY
LANACP MOP Downline Load Service
Found LAN device EZA0, hardware address 08-00-2B-30-8D-1C
```

```
%%%%%%%%%% OPCOM 10-JAN-2001 06:47:35.25 %%%%%%%%%%%
Message from user SYSTEM on GALAXY
LANACP MOP Downline Load Service
Found LAN device EZB0, hardware address 08-00-2B-30-8D-1D
```

```
%%%%%%%%%% OPCOM 10-JAN-2001 06:47:54.80 %%%%%%%%%%%
Message from user SYSTEM on GALAXY
LANACP MOP V3 Downline Load Service
Volunteered to load request on EZA0 from ZAPNOT
Requested file: $64$DIA24:<SYS11.>[SYSCOMMON.SYSEXEXE]APB.EXE
```

```
%%%%%%%%%% OPCOM 10-JAN-2001 06:48:02.38 %%%%%%%%%%%
Message from user SYSTEM on GALAXY
LANACP MOP V3 Downline Load Service
Load succeeded for ZAPNOT on EZA0
System image, $64$DIA24:<SYS11.>[SYSCOMMON.SYSEXEXE]APB.EXE (Alpha image)
```

LAN\$ACP.LOG ファイルの内容を次に示します。

```
10-JAN-2001 06:47:35.02 Found LAN device EZA0, hardware address
08-00-2B-30-8D-1C
10-JAN-2001 06:47:35.18 Found LAN device EZB0, hardware address
08-00-2B-30-8D-1D
10-JAN-2001 06:47:35.25 LANACP initialization complete
10-JAN-2001 06:47:45.39 Enabled LAN device EZA0 for MOP downline load service in
exclusive mode
10-JAN-2001 06:47:54.70 Volunteered to load request on EZA0 from ZAPNOT
Requested file: $64$DIA24:<SYS11.>[SYSCOMMON.SYSEXEXE]APB.EXE
10-JAN-2001 06:48:02.23 Load succeeded for ZAPNOT on EZA0
MOP V3 format, System image, $64$DIA24:<SYS11.>[SYSCOMMON.SYSEXEXE]APB.EXE
Packets: 2063 sent, 2063 received
Bytes: 519416 sent, 4126 received, 507038 loaded
Elapsed time: 00:00:07.42, 68276 bytes/second
```

10.8.5 クロス・アーキテクチャ・ブート

LANの機能拡張によって、OpenVMS Cluster システム内でのクロス・アーキテクチャ・ブートが可能になりました。VAX ブート・ノードは、Alpha サテライトに対してブート・サービスを提供することができ、Alpha ブート・ノードはVAX サテライトに対してブート・サービスを提供することができます。なお、各アーキテクチャとも、インストールおよび更新に使用するシステム・ディスクを持つ必要があることに注意してください。

10.9 LAN MOP ダウンライン・ロード・サービスの管理

LANACP LAN サーバ・プロセスは、LAN 運用時ノード・データベースおよびデバイス・データベースを保守します。LANCP ユーティリティには、次の機能を持つコマンドが用意されています。

- MOP ダウンライン・ロード状態およびカウンタ情報を表示する。
- カウンタ情報をクリアする。
- OPCOM メッセージおよびパケット・トレースを許可または禁止する。

カウンタおよび状態情報は、ノードおよびデバイスごとに保守されます。カウンタ情報には、送受信されたバイトおよびパケット数、送信エラー、プロトコル違反やタイムアウトなどの論理エラー、ロード要求数が含まれます。状態情報には、最終ロード時刻および最終ロード状態が含まれます。

10.9.1 MOP ダウンライン・ロード・サービスの許可

MOP ダウンライン・ロード・サービスを許可するには、次の形式で SET DEVICE コマンドを入力します。

```
SET DEVICE デバイス名/DLL=ENABLE
```

このコマンドでは、**デバイス名**パラメータで LAN コントローラのデバイス名を指定します。

このコマンドについての詳細は、10.6.2 項「LAN デバイス・データベースへのデバイスの入力」を参照してください。

10.9.2 MOP ダウンライン・ロード・サービスの禁止

MOP ダウンライン・ロード・サービスを禁止するには、次の形式で SET DEVICE コマンドを入力します。

```
SET DEVICE デバイス名/DLL=DISABLE
```

このコマンドでは、**デバイス名**パラメータで LAN コントローラのデバイス名を指定します。

このコマンドについての詳細は、10.6.2 項「LAN デバイス・データベースへのデバイスの入力」を参照してください。

10.9.3 状態データとカウンタ・データの表示

MOP ダウンライン・ロード状態を表示するには、次の形式で SHOW DLL コマンドを入力します。

```
SHOW DLL
```

次の例は、特定のノードについてのカウンタ情報を示しています。

```
LAN MOP DLL Status:
```

```
EXA enabled in exclusive mode for known nodes only, data size 1482 bytes  
FXA disabled
```

	#Loads	Packets	Bytes	Last load time	Last loaded
EXA	5	1675	4400620	10-JAN-2004 10:27.51	GALAXY
FXA	0	0	0		

このノードには EXA (DEMNA) および FXA (DEMFA) という 2 つの LAN デバイスがあります。MOP ダウンライン・ロード・サービスは、EXA 上で排他的モードで許可されます。

LANACP ノード・データベースで定義されているノードに限って、要求が応答されます。ロード・メッセージ内のイメージ・データのサイズは 1482 バイトです。5 つのダウンライン・ロードがあり、最後のダウンライン・ロードは、ノード GALAXY 上で 10:27 に起こりました。最終的に、FXA に対してダウンライン・ロードは記録されません。これは、ダウンライン・ロード・サービスが現在禁止されているためです。

LAN\$ACP.LOG ファイルに記録されている最新のダウンライン・ロード処理を表示するには、次の形式で SHOW LOG コマンドを入力します。

```
SHOW LOG
```

10.9.4 個々のノードの状態とカウンタ・データの表示

LAN パーマネント・ノード・データベース内のノードについて、MOP ダウンライン・ロード情報を表示するには、次の形式で LIST NODE コマンドを入力します。

```
LIST NODE ノード名 [/修飾子,...]
```

LAN 運用時ノード・データベース内のノードについて、MOP ダウンライン・ロード状態とカウンタ情報を表示するには、次の形式で SHOW NODE コマンドを入力します。

```
SHOW NODE ノード名 [/修飾子,...]
```

表 10-17 「LIST NODE および SHOW NODE コマンド修飾子」に、LIST NODE および SHOW NODE コマンド修飾子について簡単に説明します。

表 10-17 LIST NODE および SHOW NODE コマンド修飾子

修飾子	説明
/ALL	データベース内の全ノードについて情報を表示する。
/OUTPUT= ファイル名	指定されたファイルに対して、出力が行われることを指示する。ファイル名拡張子が .com である場合、出力は、DEFINE NODE または SET NODE コマンドのリスト形式になる。作成されたコマンド・ファイルは、LAN ノード・データベースを作成するのに使用できる。
/TOTAL (SHOW NODE コマンドのみ)	カウンタ合計だけを表示する。

例

次の例は、3 つのノード (GALAXY, ZAPNOT, CALPAL) が定義されているローカル・ノードで発行されたコマンドからの出力を示します。CALPAL は 2 つのロード要求を出しています。

- 最初の要求は、ローカル・ノードが自発的に受け入れた CALPAL からのマルチキャスト要求。
- 2 番目の要求は、実際のロード・データについて、CALPAL がローカル・ノードに直接送信したロード要求。2 番目のロード要求からロード完了までの経過時間は 6.65 秒。

Node Listing:

```
GALAXY (08-00-2B-2C-51-28):
  MOP DLL:  Load file:  APB.EXE
            Load root:  $64$DIA24:<SYS11.>
            Boot type:  Alpha satellite

ZAPNOT (08-00-2B-18-7E-33):
  MOP DLL:  Load file:  NISCS_LOAD.EXE
            Load root:  LAVC$SYSDEVICE:<SYS10.>
            Boot type:  VAX satellite

CALPAL (08-00-2B-08-9F-4C):
  MOP DLL:  Load file:  READ_ADDR.SYS
            Last file:  LAN$DLL:APB_X5WN.SYS
            Boot type:  Other
            2 loads requested, 1 volunteered
            1 succeeded, 0 failed
            Last request was for a system image, in MOP V4 format
```

Last load initiated 10-jan-2001 09:11:17 on EXA0 for 00:00:06.65
527665 bytes, 4161 packets, 0 transmit failures

Unnamed (00-00-00-00-00-00):

Totals:

Requests received	2
Requests volunteered	1
Successful loads	1
Failed loads	0
Packets sent	2080
Packets received	2081
Bytes sent	523481
Bytes received	4184
Last load	CALPAL at 10-jan-2004 09:11:17.29

10.9.5 カウンタ・データのクリア

全ノードおよびデバイスで MOP ダウンライン・ロード・カウンタをクリアするには、次の形式で CLEAR DLL コマンドを入力します。

CLEAR DLL

10.9.6 OPCOM メッセージ

省略時の設定では、OPCOM メッセージが許可されています。メッセージは、デバイス状態が変化したとき、ロード要求を受信したとき、およびロードが完了したとき、LANACP LAN サーバ・プロセスによって生成されます。これらのメッセージは、オペレータのコンソールに表示され、LANACP によって作成されるログ・ファイル SYS\$MANAGER:LAN\$ACP.LOG に書き込まれます。

OPCOM メッセージを許可するには、次の形式で SET ACP/OPCOM コマンドを入力します。

SET ACP/OPCOM

10.9.7 ロード・トレース機能

ロード要求について LANACP LAN サーバ・プロセスが出力するエラー・データでは、ロードの失敗の理由が十分に判断できない場合、サーバ・プロセスに対して、トレース・データの記録を指示することができます。データは、サーバによって行われる送受信ごとの、送受信パケット情報で構成され、ロードのたびにログ・ファイルに書き込まれます。ログ・ファイル名は SYS\$MANAGER:LAN\$**nodename**.LOG です。全パケット・データ、または各パケットの先頭の 32 バイトだけ、のいずれかを記録できます。

一般的なロードの流れを次に示します。

1. Program Request メッセージを、Load Assistance Multicast Address 上で、要求ノードから受信する。コード 8。
2. Assistance Volunteer メッセージを要求ノードへ送信する。コード 3。
3. Program Request メッセージを、自分のノード・アドレス上で、要求ノードから受信する。コード 8。
4. Memory Load メッセージを、要求ノードへシーケンス番号ゼロで送信する。コード 2。
5. 次のシーケンス番号 (モジュロ 256) を要求する Request Memory Load メッセージを受信する。コード 10 (10 進数)。
6. 送信するデータがなくなるまで、ステップ 4 および 5 を繰り返す。
7. Memory or Parameter Load with Transfer Address メッセージを送信する。コード 0 または 20 (10 進数)。
8. 最後のメッセージが受信されたことを示す次のシーケンス番号 (モジュロ 256) を要求する、最終 Request Memory Load メッセージを受信する。コード 10 (10 進数)。

クラスタ・サテライト・ロードの場合、最後の Memory Load メッセージには、クラスタ・パラメータが含まれます。このメッセージ、および最後の Load with Transfer Address メッセー

ジは、部分的なトレース・エコーだけが許可されている場合であっても、すべて表示されま
す。

パケット・データの部分トレースを許可するには、次の形式で SET ACP/ECHO コマンドを入
力します。

```
SET ACP/ECHO
```

パケット・データの完全トレースを許可するには、/FULL 修飾子を追加します。

```
SET ACP/ECHO/FULL
```

10.9.8 MOP コンソール・キャリア

コンソール・キャリアは、MOP コンソール・キャリア・プロトコルを使用して 管理インタ
フェースを実現している、ターミナル・サーバなどの LAN デバイスに接続するメカニズムを
備えています。LANCP ユーティリティは、この機能を CONNECT NODE コマンドの形式で
提供します。

コマンド形式は次のとおりです。

```
CONNECT NODE ノード指定 [/修飾子,...]
```

表 10-18 「CONNECT NODE コマンド修飾子」に、CONNECT NODE コマンド修飾子につ
いて、簡単に説明します。

表 10-18 CONNECT NODE コマンド修飾子

修飾子	説明
/DEVICE= デバイス名	接続に使用する LAN コントローラのデバイス名を指定する。
/DISCONNECT= 切断文字	遠隔ノードとの接続終了に使用できる文字を指定する。
/PASSWORD=16 桁の 16 進数	接続を開始するときを使用されるパスワードを指定する。
/V3 または /V4	それぞれ MOP バージョン 3 またはバージョン 4 で書式化されたメッセ ージを使用して接続を行うように指定する。

例

1. CONNECT NODE GALAXY/DEVICE=EWA0

このコマンドは、イーサネットのデバイス EWA0 を使用して、ノード GALAXY へのコン
ソール・キャリア接続を試みる。

2. CONNECT NODE 08-00-2B-11-22-33/DEVICE=EWA0/PASSWORD=0123456789ABCDEF

このコマンドは、イーサネット・デバイス EWA0 を使用し、パスワードを指定して、任
意のノード・アドレスへのコンソール・キャリア接続を試みる。

10.9.9 MOP トリガ・ブート

システムによっては、MOP 遠隔ブート要求を認識して、応答します。これらのシステムは通
常、不要なブート要求によってシステムのリポートが起動されるのを防ぐために、パスワード
または他のメカニズムを必要とします。LANCP ユーティリティは、この機能を TRIGGER NODE
コマンドの形式で実現しています。

LAN システムのリポートを要求するには、次の形式で TRIGGER NODE コマンドを入力しま
す。

```
TRIGGER NODE ノード指定 [/修飾子,...]
```

表 10-19 「TRIGGER NODE コマンド修飾子」に、TRIGGER NODE コマンド修飾子につ
いて簡単に説明します。

表 10-19 TRIGGER NODE コマンド修飾子

修飾子	定義
/DEVICE= デバイス名	ブート・メッセージを送信するのに使用する LAN コントローラのデバイス名を指定する。
/PASSWORD=16 桁の 16 進数	接続を開始するとき使用されるパスワードを指定する。

MOPバージョン 3 または 4 を送信するための形式の指定に代わって、LANCP ユーティリティは、ターゲット・ノードに対して、各形式でメッセージを 1 つ送信します。

次の例は、TRIGGER NODE コマンドの使い方を示しています。

例

1. TRIGGER NODE GALAXY/DEVICE=EWA0

このコマンドは、イーサネット・デバイス EWA0 を使用して、ノード GALAXY に対して、MOP トリガ・ブート・メッセージを送信します。

2. TRIGGER NODE 08-00-2B-11-22-33/DEVICE=EWA0/PASSWORD=0123456789ABCDEF

このコマンドは、イーサネット・デバイス EWA0 を使用し、パスワードを指定して、指定ノード・アドレスに対して、MOP トリガ・ブート・メッセージを送信します。

10.10 LAN フェイルオーバーについて

LAN フェイルオーバーは、システムを LAN デバイスの障害から保護するメカニズムです。LAN フェイルオーバーでは、複数の LAN デバイスを、LAN フェイルオーバー・セットという 1 つの仮想インタフェースに統合することによって、この保護を実現します。

フェイルオーバー・セットは、1 個のアクティブな LAN デバイスと、いくつかのアクティブでない LAN デバイスからなります。アクティブなアダプタに障害が発生すると、送受信動作は、自動的および透過的に、アクティブでないデバイスのいずれかに移されます。LAN デバイスが 1 個だけのフェイルオーバー・セットを作成することもできますが、そのセットにデバイスが追加されるまでは、フェイルオーバーは発生しません。

ソフトウェア: 論理 LAN ドライバ

論理 LAN ドライバ SYS\$LLDRIVER.EXE は、LAN フェイルオーバー・セットの動作を管理し、アクティブな LAN デバイス・ドライバへ I/O 要求を導くことによって、LAN フェイルオーバーを実現します。アクティブな LAN デバイスに障害が発生すると、LLDRIVER は、LAN デバイスに割り当てられている優先順位とリンクの状態に従って選択した、他の LAN デバイスに切り替えます。

ハードウェア: 論理 LAN デバイス

論理 LAN デバイス llc0 は、LAN フェイルオーバー・セットごとに作成されます (c は、論理 LAN デバイスを一意に識別するための、ユーザ定義のアルファベットです)。このデバイスは、擬似 LAN デバイス、つまり仮想 LAN デバイスです。アプリケーションは、I/O 要求を、論理 LAN デバイスに送ります。論理 LAN ドライバは、これらの要求を、アクティブな LAN デバイスに導きます。

ソフトウェア - LAN フェイルオーバー管理

システム管理者は、LANCP ユーティリティと LANACP ユーティリティを使用して、LAN フェイルオーバー・セットを定義し作成します。LAN フェイルオーバー・セットのコンテキストは、パーマネント LAN デバイス・データベースで管理されます。システムの起動時、LANACP が再起動されると、LANACP はパーマネント・デバイス・データベースを読み取り、このデータベース内に記述されている LAN フェイルオーバー・セットの設定を行います。

システム稼働中は、フェイルオーバの状態と構成データを参照することができます。また、パーマメント・デバイス・データベースと、運用デバイス・データベース内で管理されている現在のインスタンスの両方で、フェイルオーバ・セットの特性を変更することができます。

ネットワークの構成とハードウェアの要件

LANのフェイルオーバでは、フェイルオーバ・セット内のLANデバイスが、同じ拡張LANに存在する必要があります。この要件により、任意のノード間のLAN通信を、フェイルオーバ・イベントの発生後も継続できます。

LANフェイルオーバをサポートするLANデバイスは、Alphaシステム用としてはDE500-BA、DE504-BA、DE600、およびその改良版、I64システム用としてはIntel 82559、A5230A、およびA5506B、AlphaシステムとI64システム用としてはすべてのギガビット・デバイスがあります。AlphaServer DS25の内蔵Intel 82559チップも、同様にサポートしています。

ネットワーク接続障害の検出

一般的に、LANデバイスはネットワーク・スイッチに接続されます。この接続の状態は、「リンク・アップ」または「リンク・ダウン」と表現されます。「リンク・アップ」状態とは、LANデバイスからスイッチへの有効な接続があり、LANデバイスとスイッチの間でネットワーク・データを転送できる状態です。「リンク・ダウン」状態とは、ケーブル障害や切断、または接続の片側に障害が発生しているために、有効な接続がない状態です。スイッチの電源を切るかリセットすると、スイッチが再度動作状態になるまで、リンクは「ダウン」状態になります。LANデバイスがリセットされると、LANドライバがリセットを完了し、LANデバイスが再初期化されるまで、リンクは「ダウン」状態になります。

LANフェイルオーバは、LANデバイスの障害、デバイスとスイッチを接続しているケーブルの障害、およびスイッチの障害からの保護を行います。LANフェイルオーバは、スイッチを越えた場所での障害、つまり、フェイルオーバ・セット内のLANデバイスからは見えない部分のLAN障害からの保護は行いません。

フェイルオーバ・セット内の各LANデバイスは、リンクの状態を監視し、LLDRIVERにそのリンクの状態を報告します。LLDRIVERはリンクの状態を追跡し、アクティブなLANデバイスが「リンク・ダウン」状態を報告したときに、フェイルオーバを実行します。

LAN フェイルオーバの制限

LANフェイルオーバには、次の制限があります。

- LANフェイルオーバは、クラスタ・サテライトではサポートされません。
- ポイント・ツー・ポイント接続はサポートされません。他のLANデバイスを含むポイント・ツー・ポイント構成では、LLDRIVERは、障害発生後、次のアクティブなデバイスとして同じポイント・ツー・ポイント・デバイスが選択されることは保証できません。LANデバイスが2個だけの構成でも、リンク状態のタイミングは変わりやすいため、期待どおりのフェイルオーバ・セット動作をすることは保証できません。
- LANデバイスやスイッチへの接続の範囲外で発生した障害は、検出できません。
- ジャンボ・フレームが有効になっている場合は、フェイルオーバ・セット内のすべてのLANデバイスはジャンボ・フレームをサポートしていなければならず、ネットワークのインフラストラクチャも、それに合わせて設定されていなければなりません。たとえば、あるLANデバイスから他のノードへのパスがジャンボ・フレームをサポートしている場合、フェイルオーバ・セット内の他のすべてのLANデバイスから同じノードへのパスも、ジャンボ・フレームをサポートしていなければなりません。

LAN フェイルオーバの管理

LANの運用時デバイス・データベースおよびパーマメント・デバイス・データベースには、システム上に存在する各LANデバイスのエントリが1つ含まれています。論理LANデバイスllcは、システム管理者がLANフェイルオーバ・セットを作成したときに、LANCPおよびLANACPを通してこれらのデータベースに追加されます。

LLc デバイスを LAN のパーマネント・デバイス・データベースに入れるか、既存のエントリを変更するには、次の構文で、LANCP コマンドの DEFINE DEVICE LLc を入力します。

```
DEFINE DEVICE LLc[/修飾子]
```

このコマンドで設定した内容は、以後のブート時に有効になります。LLc デバイスを LAN の運用時デバイス・データベースに入れるか、既存のフェイルオーバ・セットを変更するには、次の構文で、LANCP コマンドの SET DEVICE LLc を入力します。

```
SET DEVICE LLc[/qualifiers]
```

このコマンドで変更した内容は、すぐに有効になります。

LAN デバイスの優先順位は、論理 LAN デバイス上で管理されるのではなく、LAN デバイス上で個々に管理されます。

10.10.1 LAN フェイルオーバ・セットの作成

論理 LAN デバイスは、LAN フェイルオーバ・セットの作成時に作成されます。LAN フェイルオーバ・セットを作成するには、次の構文で、LANCP コマンドの SET DEVICE LLc を入力します。

```
SET DEVICE LLc/FAILOVER_SET=( デバイス名 [...])
```

このコマンドでは、デバイス名に、LAN 物理デバイス名を指定します。例を次に示します。

```
LANCP> SET DEVICE LLA/FAILOVER_SET=(EWA,EIA,EWB)
```

指定した LAN デバイスを使用しているユーザがいる場合、このコマンドは失敗します。LANCP コマンドの SHOW CONFIG/USERS を入力することで、アクティブなユーザを表示できます。ユーザとは、アプリケーションがデバイスに割り当てたチャンネルのことです。1つのアプリケーションで、複数のチャンネルを割り当てることができます。

また、LANCP コマンドの DEFINE DEVICE を使用して、LAN パーマネント・デバイス・データベース内にフェイルオーバ・セットを設定することもできます。システムのブートの際、ユーザ(アプリケーション)が起動される前に、フェイルオーバ・セットが初期化されます。LANACP は、停止して再起動されると、LAN パーマネント・デバイス・データベース内に定義されているフェイルオーバ・セットを設定しようとします。この場合、LANACP がフェイルオーバ・セットの設定を正しく行えるように、既存のユーザを停止しなければなりません。

10.10.2 LAN フェイルオーバ・セットへの LAN デバイスの追加

追加の LAN デバイスを指定することで、既存のフェイルオーバ・セットに LAN デバイスを追加できます。たとえば、上記のフェイルオーバ・セットに EWC を追加するには、次のコマンドを入力します。

```
LANCP> SET DEVICE LLA/FAILOVER_SET=EWC
```

10.10.3 LAN フェイルオーバ・セットからの LAN デバイスの削除

LAN フェイルオーバ・セットから LAN デバイスを削除するには、次の構文で、LANCP コマンドの SET DEVICE を入力します。

```
SET DEVICE LLc/NOFAILOVER_SET=( device_name [...])
```

このコマンドでは、削除する LAN デバイス名を指定します。例を次に示します。

```
LANCP> SET DEVICE LLA/NOFAILOVER_SET=EWB
```

指定したデバイスのいずれかがアクティブな LAN デバイスの場合、このコマンドは失敗します。

10.10.4 LAN フェイルオーバー・セットの有効化

LAN フェイルオーバー・セットを有効にすると、フェイルオーバー・セットがアクティブになります。LLDRIVER は、優先順位とリンクの状態に応じて LAN デバイスのいずれかを選択し、ユーザから論理 LAN デバイスへの I/O を可能にします。

LAN フェイルオーバー・セットを有効にするには、次の構文で、LANCP コマンドの SET DEVICE を入力します。

```
SET DEVICE llc/ENABLE
```

10.10.5 LAN フェイルオーバー・セットの無効化

LAN フェイルオーバー・セットを無効化すると、論理 LAN デバイスが非アクティブになります。論理 LAN デバイスが非アクティブになると、ユーザの I/O 要求は、エラー状態で完了します。LAN フェイルオーバー・セットを無効化すると、LAN デバイスをフェイルオーバー・セットに追加したり、フェイルオーバー・セットから削除したりできます。

LAN デバイスは、アクティブなデバイスでない限り、有効状態の LAN フェイルオーバー・セットから削除できません。LAN デバイスは、使用中のユーザがいない限り、追加することができます。

LAN フェイルオーバー・セットを無効にするには、次の構文で、LANCP コマンドの SET DEVICE を入力します。

```
SET DEVICE llc/DISABLE
```

論理 LAN デバイスを使用しているユーザがいる場合、このコマンドは失敗します。

10.10.6 LAN フェイルオーバー・デバイスの優先順位の設定

LAN フェイルオーバー・セットのアクティブな LAN デバイスを LLDRIVER が選択するときに、特定の物理 LAN デバイスを優先させるには、次の構文で、LANCP コマンドの SET DEVICE/PRIORITY を入力します。

```
SET DEVICE デバイス名/PRIORITY= 値
```

このコマンドでは、デバイス名に LAN デバイス名を指定し、値パラメータには整数を指定します。例を次に示します。

```
LANCP> SET DEVICE EIA/PRIORITY=20
```

アクティブにするメンバを選択する際に、LLDRIVER は、優先順位が最も高く、「リンク・アップ」状態のデバイスを選択します。

10.10.7 LAN フェイルオーバー・セットの packetsize の設定

LAN フェイルオーバー・セットの省略時の最大 packetsize は、標準のイーサネットの最大 packetsize である 1518 バイトです。フェイルオーバー・セット内のすべての LAN デバイスがジャンボ・フレームをサポートしているときには、ジャンボ・フレームの使用を有効にして、標準のイーサネットの最大 packetsize またはジャンボ・packetsize を選択することができます。この選択を行うには、次の構文で、LANCP コマンドの SET DEVICE/SIZE を入力します。

```
SET DEVICE llc/SIZE=STANDARD
```

```
SET DEVICE llc/SIZE=JUMBO
```

LAN_FLAGS システム・パラメータは、通常、ギガビット LAN デバイス上でジャンボ・フレームの使用を有効にするために使用されます。または、LANCP コマンドの SET DEVICE/[NO]JUMBO を使用すると、特定のデバイス上のジャンボ・フレームを有効または無効にできます。論理 LAN デバイスの packetsize 選択は、フェイルオーバー・セット内の LAN デバイスのジャンボ設定より優先されます。フェイルオーバー・セットが無効状態のときに、設定を変更することができます。

10.10.8 LAN フェイルオーバーの特性の表示

LAN フェイルオーバーの状態を表示するには、次の構文で、LANCP コマンドの SHOW DEVICE llc を入力します。

```
SHOW DEVICE llc/CHARACTERISTICS
```

特定のノードの LAN フェイルオーバーに固有の特性が表示されます。例を次に示します。

```
Device Characteristics LLA0:
```

```
Value Characteristic
```

```
.  
. .  
. .
```

```
Disabled Jumbo frames  
"EIA" Failover device (active)  
"EWA" Failover device  
Enabled/Active Logical LAN state  
0 Failover priority
```

10.10.9 LAN フェイルオーバー・カウンタの表示

LLDRIVER は、論理 LAN デバイスからアクティブな LAN デバイスに、I/O 要求をリダイレクトします。デバイス・カウンタやドライバ内部のカウンタを表示する LANCP コマンドは、アクティブな LAN デバイスにリダイレクトされます。

10.10.10 LAN フェイルオーバー・セットのチェック

LAN フェイルオーバー・セットに組み込まれたネットワーク・デバイスは、LAN フェイルオーバーを正しく動作させるために、同じローカル・ネットワーク上で、物理的に冗長なパスを構成していなければなりません。ネットワークは通常は安定しているため、LAN フェイルオーバー・セットのアクティブ・メンバは、頻繁には変わりません。ただし、アクティブなメンバが切り替わるときには、アイドル NIC が適切にセットアップされている必要があります。

システム管理者は、LANCP の修飾子 /SWITCH を使用して LAN 障害をシミュレートすることにより、フェイルオーバー・セットの各メンバをチェックすることができます。/SWITCH 修飾子は、アクティブなデバイス上のネットワーク障害をシミュレートします。そして、フェイルオーバー・セットから他のデバイスを選択して、アクティブ・デバイスとします。

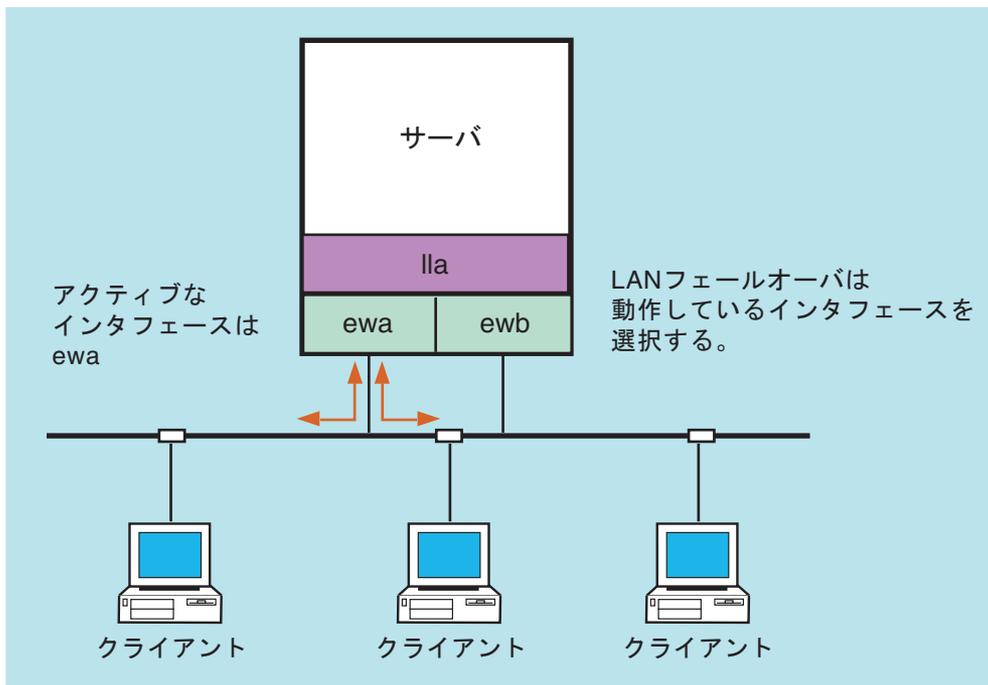
LAN 障害をシミュレートするには、次の構文で、LANCP コマンドの SET DEVICE llc を入力します。

```
SET DEVICE llc/SWITCH
```

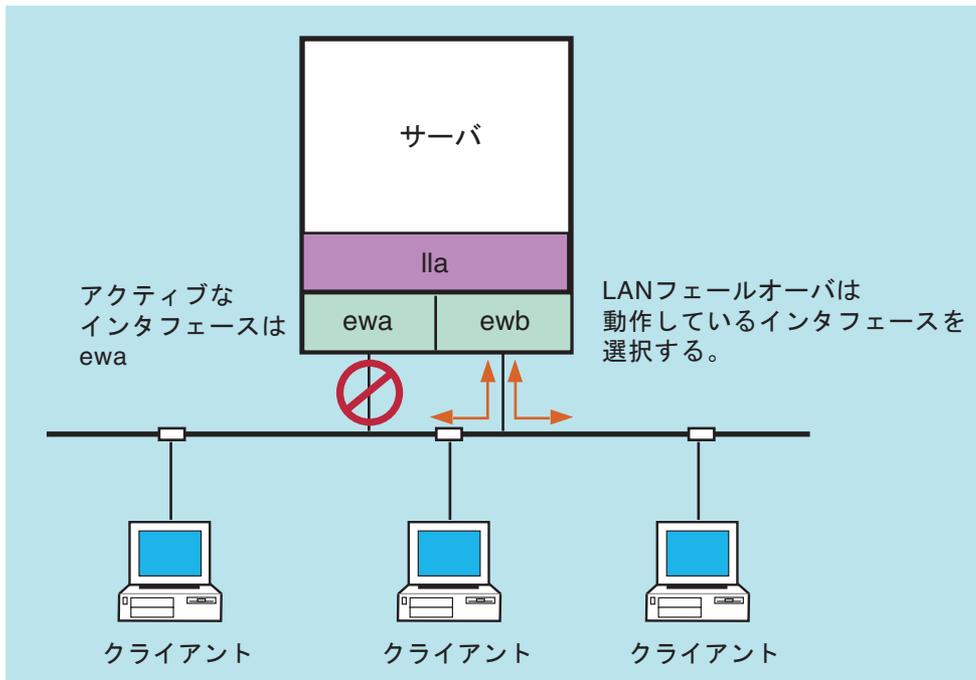
10.10.11 LAN フェイルオーバーの例

図 10-1 「LAN フェイルオーバー」に、LAN フェイルオーバーの例を示します。

図 10-1 LAN フェイルオーバー



LANフェールオーバー



VM-1105A-AI

第11章 InfoServer システムの管理

この章では、InfoServer の機能と InfoServer Client for OpenVMS ソフトウェアについて説明します。このソフトウェアは OpenVMS システムが InfoServer デバイス・サービスを利用できるようにするためのものです。また、この章では、システム上でこのクライアント・ソフトウェアを起動し、InfoServer を公用デバイスとして使用するための作業についても説明します。この章では次の作業について説明します。

作業	参照箇所
サーバ管理セッションの開始	11.3 項 「サーバ管理セッションの開始」
InfoServer Client for OpenVMS の自動起動	11.5.3 項 「InfoServer Client for OpenVMS の自動起動」
InfoServer デバイスを自動的に使用可能にする方法	11.6.3 項 「InfoServer デバイスを自動的に使用可能にする方法」

さらに、次の項目について説明します。

項目	参照箇所
InfoServer の機能	11.1 項 「InfoServer の機能について」
LASTport プロトコル	11.2 項 「LASTport プロトコルについて」
InfoServer Client for OpenVMS の機能	11.4 項 「InfoServer Client for OpenVMS 機能について」
LASTCP ユーティリティの機能	11.5 項 「LASTCP ユーティリティの機能について」
LADCP ユーティリティの機能	11.6 項 「LADCP ユーティリティの機能について」

11.1 InfoServer の機能について

InfoServer システムは**仮想デバイス・サーバ**です。ローカル・エリア・ネットワーク (LAN) 上のクライアント・システムは、これにより、コンパクト・ディスク、読み書き可能ディスク、光磁気 (MO) デバイス、およびテープを利用することができます。InfoServer クライアント・ソフトウェアを実行しているシステムであれば、InfoServer システムが提供する仮想デバイスに接続して、ローカルに接続されたデバイスであるかのようにそれらを使用することができます。

ファイル・サーバと異なり、InfoServer システムは、仮想デバイス上のファイル・システムを意識しません。つまり、InfoServer システムでは、どのようなオンディスク構造を持つディスクでも利用できるということです。クライアント・システム自体がオンディスク構造を解釈し、各クライアントは独自のファイル・システムを使用してデータにアクセスすることができます。1つの InfoServer システムで、複数のオンディスク構造を同時にサポートし、同時にアクセスすることができます。

InfoServer システムの機能を次に示します。

- コンパクト・ディスクを使用できるようにする

InfoServer システムでは、サーバのブート時、またはコンパクト・ディスクが InfoServer のドライブに挿入されたときに、自動的にコンパクト・ディスクが使用できるようになる。このとき、そのボリューム・ラベルをサービス名として使用する。システム管理者が特別な処置を行う必要はない。クライアント・システムがコンパクト・ディスクにバインドし、それをボリューム・ラベルのもとでマウントする。

InfoServer システムでは、OpenVMS クライアントは ODS-2 形式のコンパクト・ディスクを自動的に使用できるようになっている。High Sierra および ISO-9660 形式のコンパク

ト・ディスク，およびその他のタイプの媒体は，InfoServer の管理インタフェースを使って手動で認識させることができる。

- SCSI テープを使用できるようにする

InfoServer システムでは，サービス名を使用して，ネットワークで SCSI テープ・デバイスを使用することができる。クライアント・システムはこれらのテープ・デバイスに接続し，ローカルにデバイスとして使用することができる。

- 読み書き可能ディスク・パーティションを提供する

パーティションとは，InfoServer 読み書き可能ディスクの論理サブセットである。1 つのディスクをいくつかのパーティションに分割し，それぞれを個別にネットワークで使用することができる。リモート・クライアント・システム側からは，各パーティションが 1 つのディスクとみなされる。たとえば，InfoServer Client for OpenVMS を使用しているクライアント・システムは，ローカル・ハード・ディスクと同様にパーティションにアクセスし使用することができる。

- OpenVMS システムの初期ロード・システムとして機能する

InfoServer システムは，保守操作プロトコル (MOP) 要求に応答することにより，1 次ブートストラップ・プログラムを OpenVMS システムにダウンライン・ロードすることができる。サーバは，MOP ダウンライン・ロード・ファイルを OpenVMS ソフトウェア・ディストリビューション・コンパクト・ディスク上で探すことができ，InfoServer 形式でフォーマットされた読み書き可能ディスク上の一時 MOP パーティションにコピーする。

初期システム・ロード (ISL) ブートストラップ・プログラムは，ソフトウェア・ディストリビューション・コンパクト・ディスクに接続し，スタンドアロン BACKUP をブートする。次に BACKUP ユーティリティにより，OpenVMS オペレーティング・システムのセーブ・セットがコンパクト・ディスクからシステムに接続されている読み書き可能ディスクにコピーされる。以降の OpenVMS のブートはすべてこのローカルの読み書き可能ディスクから行われる。

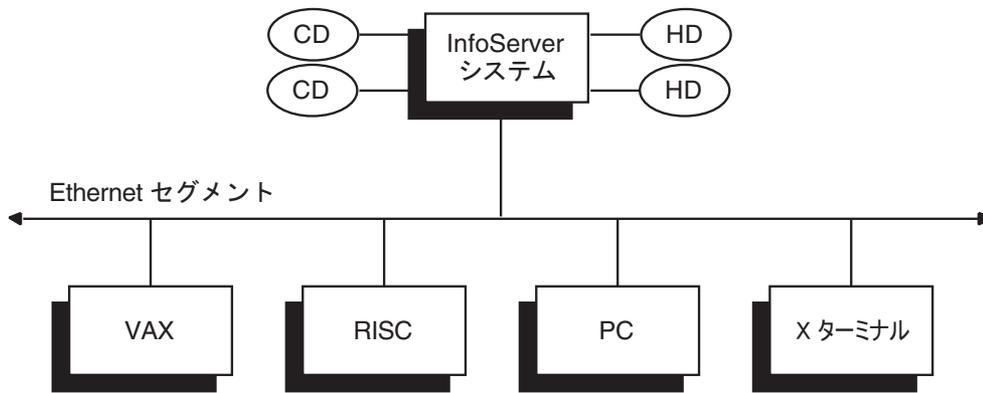
初期システム・ロード (ISL) は，164 システムではサポートされていない。164 コンソールは，保守操作プロトコル (MOP) をサポートしていない。ISL をサポートする代替の手段が，将来のリリースの OpenVMS に追加される。164 での初期ロードは，既存の InfoServer ハードウェアからはサポートされない。

- その他の製品のダウンライン・ロード

InfoServer システムを使用して，任意のイーサネット製品をファイル名でロードすることができる。すなわち，サーバは必要なファイルを見つけるために，ネットワーク制御プログラム (NCP) データベース・エントリを必要としない。たとえば，X ターミナル・クライアントは，InfoServer システムを使用して，システム・ソフトウェアをダウンライン・ロードする。特殊な MOP パーティションを作成し，必要なファイルをそのパーティションにコピーすることができる。また，サーバは，イーサネット・アドレスによるサービスのダウンライン・ロードをサポートする。各 InfoServer システムは，それぞれ最高 100 種類のダウンライン・ロードを同時に処理することができる。これは，ロードする時にプロセスを起動しなければならないホスト・ベースのダウンライン・ロードよりも効率的である。

図 11-1 「InfoServer システムによるクライアントへのサービスの提供」に，InfoServer システムとクライアント・システムとの関係を示します。この図では，サーバに接続されている 2 つのコンパクト・ディスクと 2 つのハード・ディスクがクライアント・システムからはローカル・デバイスとして認識されています。VAX システムと RISC ワークステーションは，ソフトウェアの配布とオンライン・ドキュメンテーション用に 1 つまたは 2 つのコンパクト・ディスクを使用し，一方 PC は，InfoServer システム上のディスク・パーティションを参照することができます。また，X ターミナルは InfoServer システムからブートし，ページ・ファイル，フォント・ファイル，カスタマイズ・ファイル用に InfoServer ディスクを使用します。

図 11-1 InfoServer システムによるクライアントへのサービスの提供



ZK5901AGE

InfoServer システムは、イーサネット LAN に接続してシステムをオンにすれば使用することができます。サーバが初期化 (**ブートストラップ**) されると、サーバ・ソフトウェアにより、クライアント・システムは接続されたデバイス媒体を自動的に使用できるようになります。このことをサーバ・ソフトウェアによる **サーブ** といいます。サーバのドライブにコンパクト・ディスクを挿入すると、サーバが新しいデバイスを検出して、ボリューム・ラベルをサービス名として使用することにより、クライアント・システムにこの新しいデバイスを自動的にサーブします。

サーバは、あらかじめ InfoServer ソフトウェアがインストールされている内部読み書き可能デバイスからブートストラップします。InfoServer ソフトウェアの更新内容は、コンパクト・ディスクで配布されます。更新用コンパクト・ディスクを入手したら、以降のブート用に新しいソフトウェアを内部デバイスにインストールします。InfoServer ソフトウェアをコンパクト・ディスクから更新するには、次の手順に従います。

1. InfoServer に接続されたコンパクト・ディスク・ドライブにディスクを挿入する。
2. InfoServer ソフトウェアを内部読み書き可能デバイスに移す。InfoServer プロンプトに対して、次の形式でコマンドを入力する。ただし **n** はドライブ番号である。

InfoServer 100 または InfoServer 150 システムの場合。

```
InfoServer> UPDATE SYSTEM DKn:
```

InfoServer 1000 システムの場合。

```
InfoServer> UPDATE SYSTEM DKn: FLASH
```

次に InfoServer システムをブートするときには、更新されたソフトウェアが実行されます。

InfoServer ソフトウェアを更新するには、Software Products Library (以前は ConDIST と呼んでいました) を使用します。InfoServer システムにログインした後、次の手順を実行してください。

1. [INFOSERVxxx] ディレクトリ構造を格納したディスクを、InfoServer システムに接続されたコンパクト・ディスク・ドライブに挿入する。
2. InfoServer> プロンプトに対して、次の形式でコマンドを入力する。ただし、**n** はドライブ番号である。
 - InfoServer 100 または InfoServer 150 システムの場合は、次の形式でコマンドを入力する。
UPDATE SYSTEM DKn:
 - InfoServer 1000 システムの場合は、次の形式でコマンドを入力する。
UPDATE SYSTEM DKn: FLASH

これらのコマンドは、InfoServer ソフトウェアを内蔵の読み込み/書き込みデバイスに移動させる。次回 InfoServer システムをブートすると、更新されたソフトウェアが実行される。また、Software Products Library ディスクからサーバをブートすることもできる。

サーバの機能をカスタマイズしたい場合は、サーバにログインし、サーバの各種コマンドを入力することにより、InfoServer の機能を制御することができます。詳細は『InfoServer System Operations Guide』を参照してください。

11.1.1 複数のサーバの自動サービスについて

InfoServer システムは、サーバの電源が最初に投入されたとき、あるいは取り外し可能デバイス(コンパクト・ディスクなど)がドライブに挿入されたときに、ローカルに接続されたデバイスをクライアントに自動的にサーブします。サーバは各デバイスのボリューム・ラベルを読み取って、そのラベルをクライアントに提供するサービスの名前として使用します。



注意:

自動サービス機能は、InfoServer コマンド SET SERVER AUTOMOUNT を使用して使用禁止にすることができます。

複数のサーバが同じサービスを提供している場合、クライアントはレーティング方式に従って適切なサービスを選択します。詳細は『InfoServer System Operations Guide』の CREATE SERVICE コマンドの説明を参照してください。

コンパクト・ディスクをサーバのディスク・ドライブから取り出すと、InfoServer システムは、クライアントから関連するサービスへのすべての接続を切断します。また、クライアント・システムへの関連するサービスの提供も停止します。

11.1.2 サービスの中断を少なくする高可用性機能

InfoServer システムには、OpenVMS クライアントに特に有用な可用性の高い機能があります。サーバが何らかの理由で(サーバがリブートされた場合やユーザがコンパクト・ディスクを取り出した場合など)サービス接続を切断した場合、OpenVMS クライアントは、そのボリュームに対してマウント・チェックを行います。同じサービスが LAN 上の別の InfoServer システムで提供されている場合、そのクライアントは自動的にそのサービスに接続します。

たとえば、2つのサーバ上のそれぞれのドライブに同じ OpenVMS オンライン・ドキュメンテーションが格納されたコンパクト・ディスクが装着されている場合を考えてください。1つのサーバまたはドライブが故障した場合は、もう1つのサーバ上のディスクへの接続が新たに確立されます。これにより、ファイル操作は正常に続けられ、ユーザはサービス中断をほとんど意識しないですみます。

11.1.3 X ターミナル・クライアントのサポート

X ターミナル・クライアントは、InfoServer システムを使用して、システム・ソフトウェアのダウンロード、フォント・サービスの準備、構成情報の保存、InfoServer ディスクとの間のメモリのページ処理を行います。たとえば、弊社の VXT 2000 ウィンドウ・ターミナル用システム・ファイルは、InfoServer システム上のコンパクト・ディスクからインストールすることができます。これらのファイルをインストールしておくと、各ターミナルに電源が投入されたとき、要求があるとすぐにダウンライン・ロードされます。

このターミナルでは必要に応じて、InfoServer ディスク上に動的にパーティションを割り当てるすることができます。たとえば、ユーザがターミナルのカスタマイズ情報を保存するよう要求すると、InfoServer システムにより自動的にディスク・パーティションが作成され、その情報を格納し、さらにそのパーティションにネットワーク・サービス名が付けられます。カスタマイズ情報が保存されると、ユーザはいつでもその情報を呼び出すことができます。

InfoServer クライアントである VXT 2000 ターミナルは、仮想メモリ・マシンとして機能することもできます。このようなターミナルでは、必要に応じてメイン・メモリのセクションを InfoServer ディスクとの間でページングすることができます。VXT クライアントにはローカ

ル・ディスクがないため、InfoServer ディスクがページ・ディスクとして使用されます。メイン・メモリがディスクにページ・アウトされると、VXT クライアントはパーティションを作成するよう InfoServer システムに要求します。このパーティションは必要に応じて自動的に拡張されます。パーティションとそのネットワーク・サービス名は動的に作成されます。このとき、ユーザによる処置は必要ありません。

省略時の設定では、InfoServer ディスク DK1 (各 InfoServer 150 システムとともに出荷される内部ディスク) に対して、VXT 2000 クライアントがパーティションを遠隔割り当てできるようになっています。InfoServer のコマンドを使用すれば、他のディスクも使用することができます。

11.2 LASTport プロトコルについて

InfoServer システムは、LASTport トランスポート・プロトコルと LASTport/Disk および LASTport/Tape というシステム・アプリケーション・プロトコルを使用して、LAN 上で仮想デバイスを利用できるようにします。これらのプロトコルにより、ディスクやテープ・デバイスへの高性能のアクセスを行うことができます。InfoServer システムは、このプロトコルのサーバ部分を実装し、InfoServer の記憶デバイスにアクセスするクライアント・システムがクライアント部分を実装します。

LASTport トランスポートを実行する OpenVMS システムでは、イーサネット・デバイスはすべて、デバイスをアクティブなネットワークに接続するか、または適切なターミネータを使ってターミネートさせる必要があります。デバイスが適切にターミネートされていないと、システムがクラッシュします。

11.2.1 LASTport トランスポート・プロトコル

LASTport プロトコルは、多くのクライアントが InfoServer システムにアクセスし、信頼のおけるトランザクションを実行できるようにする、特殊な LAN トランスポート・プロトコルです。InfoServer システムの場合、トランザクションとはデバイスの読み込み操作または書き込み操作を意味します。LASTport プロトコルにより、多くのクライアント・システムは、同時に InfoServer の記憶デバイスとの間で情報の読み込みと書き込みを行うことができます。

LASTport プロトコルは、タイマ・ベースのプロトコルではなく、トランザクション指向のプロトコルです。通常、クライアントがトランザクションを開始しない限り、クライアントと InfoServer システムとの間で情報の受け渡しは行われません。クライアント・システムはトランザクションの起動と同時にタイマを実行します。通常、2 秒から 5 秒たつと、そのトランザクションが失われたとみなし、トランザクションを再度実行します。

LASTport プロトコルにはルーティング機能はなく、LAN 内でのみ実行されます。LASTport プロトコルのタイプは 80-41 です。拡張 LAN が何らかのフィルタリング・デバイスを使用する場合は、このプロトコル・タイプにフィルタリングを行わないようにして、クライアントがフィルタリング・デバイスを通して InfoServer にアクセスできるようにする必要があります。

InfoServer システムは、LASTport プロトコルのマルチキャスト・アドレス機能を使用してデバイスとの接続を確立します。マルチキャスト・アドレスの形式は 09-00-2B-04-**nn-nn** (**nn** は使用可能になっている作業グループによって決まる) です (『InfoServer System Operations Guide』を参照)。

11.2.2 LASTport/Disk プロトコル

LASTport/Disk プロトコルは、LASTport トランスポートを使用する特別なデバイス・プロトコルです。すなわち、LASTport/Disk メッセージは LASTport メッセージ内に表示されます。LASTport/Disk プロトコルには、基礎となるどのファイル・システムからも独立して論理ブロックの読み込みと書き込みを行うためのメカニズムがあります。LASTport/Disk プロトコルを実装したクライアントは、ローカルでファイル・システムを解釈します。LASTport/Disk プロトコルを使用してコンパクト・ディスクと読み書き可能ディスクへアクセスすると、InfoServer システムは複数のクライアントのオペレーティング・システムとオンディスク構造を同時にサポートすることができます。

LASTport/Disk プロトコルには、コンパクト・ディスクと読み書き可能ディスクにアクセスするための命名機能もあります。InfoServer システムは、各仮想ディスクにサービス名を割り当て、クライアントがそれらの名前で LAN を照会できるようにします。要求されたサービスが見つかったら、クライアントはそのサービスに接続し、デバイス・アクセスを開始します。同じサービス名で同じ仮想ディスクを使用できる場合に、使用可能なデバイス間で負荷分散を行う機能もあります。

11.2.3 LASTport/Tape プロトコル

LASTport/Disk プロトコルと同様、LASTport/Tape プロトコルも LASTport トランスポートを使用します。すなわち、LASTport/Tape メッセージは LASTport メッセージ内に表示されます。LASTport/Tape プロトコルには、テープ・レコードの読み込みと書き込みを行うメカニズムがあります。InfoServer システムに接続されたテープ・デバイスは、テープ・クライアントではローカルに接続されているデバイスとして認識されます。

LASTport/Tape プロトコルには、テープにアクセスするための命名機能もあります。InfoServer システムは各テープ・デバイスにサービス名を割り当て、クライアントがその名前で LAN を照会できるようにします。要求されたサービスが見つかったら、クライアントはそのサービスに接続し、テープ・アクセスを開始します。

11.3 サーバ管理セッションの開始

サーバ管理セッションは、次のように、ローカル・コンソール・ターミナルからでもリモート・コンソール・ターミナルからでも開始することができます。

- **ローカル・セッションの場合**は、VT100 ANSI のエスケープ・シーケンスを解釈できるターミナルを InfoServer システム・ユニットの裏側のシリアル・ポート (InfoServer 150 ユニット上の MMJ1) に接続する。ターミナルの設定は、9600 ボー、8 ビット、パリティなし。
- **遠隔セッションの場合**は、ローカル・エリア・ターミナル (LAT) サーバを介して、InfoServer システムに接続する。

多くのネットワーク・サーバと同様、InfoServer システムは、LAT サービスをその管理インタフェースに対して宣言し、ターミナル・サーバに接続されている遠隔ターミナルからの接続を受け入れる。したがって、拡張 LAN 上のターミナル・サーバに接続されているすべてのターミナルは、InfoServer システムのコンソール・ターミナルとして動作することができる (ただし、ユーザが InfoServer 管理パスワードを知っている場合)。

サーバの省略時のサービス名の決定

初めて InfoServer システムへの遠隔接続を確立する場合は、サーバの省略時の名前を決定する必要があります。この名前は、InfoServer システムのキャビネット上の 16 進のイーサネット・データリンク・アドレスに、LAD_ という 4 文字の接頭辞を付加することにより決定します。この省略時の名前は、InfoServer の SET SERVER NAME コマンドを使って変更することができます。

サーバの名前は、接続先の LAT サービス名です。たとえば、省略時のサーバ名は、LAD_08002B15009F です。InfoServer システムを管理する場合は、ターミナル・サーバのプロンプトに次のコマンドを入力します。

```
Local> CONNECT LAD_08002B15009F
```

LAT サービスの接続の開始についての詳細は、使用しているターミナル・サーバのユーザ・ガイドを参照してください。

InfoServer パスワードの入力

InfoServer システムに接続した後、管理セッションを開始するには、InfoServer パスワードが必要です。省略時のサーバ・パスワードは ESS です。このパスワードは、InfoServer の SET SERVER PASSWORD を使用して変更することができます。

例

次の例では、DECserver 500 ターミナル・サーバを使用して、セッションを開始しています。

```
Local> CONNECT LAD_08002B133C1C
Password: ESS (not echoed)
Local -010- Session 1 to LAD_08002B133C1C established
```

```
DEC InfoServer V3.1
```

```
InfoServer> SHOW SERVER
```

この例において、ターミナル・サーバのプロンプトは Local> です。ここからサービス名が LAD_08002B133C1C の InfoServer システムへの LAT セッションを開始しています。次に、InfoServer システムからサーバ・パスワードの入力を求めるプロンプトが表示されます。正しいパスワードを入力すると、InfoServer> プロンプトが表示されて、InfoServer のコマンドを入力することができるようになります。

セッションの終了

管理セッションを終了するには、InfoServer> プロンプトに対して EXIT コマンドを入力します。LAT 接続を介して管理セッションが行われている場合は、EXIT コマンドを実行するとターミナル・サーバの Local> プロンプトに戻ります。

11.3.1 サーバ管理コマンド

表 11-1 「InfoServer コマンド」に、InfoServer コマンドについてまとめます。

表 11-1 InfoServer コマンド

コマンド	機能
BACKUP	InfoServer 形式のディスクを保存する。
BIND	指定した ODS-2 サービスへの接続を確立し、そのサービス用に仮想デバイス VDK1 を作成する。
CLEAR	コンソール・ターミナルの画面を消去する。
COPY	1 つのディスクまたはパーティションから、別のディスクまたはパーティションへデータをコピーする。
CRASH	サーバ・ソフトウェアに認識可能なバグチェックを行わせ、クラッシュダンプ処理が可能であればダンプを作成する。
CREATE	新しいパーティションを作成する。あるいは新しいサービスを作成する。
DELETE	以前に作成されたパーティションまたはサービスを削除する。
DISCONNECT	LASTport または LAT ターミナル・サーバ・セッションを終了する。
ERASE	指定したディスクまたはパーティションを消去する。FUNCTIONS または SERVICES データを、不揮発性ランダム・アクセス・メモリ (NVRAM) から消去する。
EXIT	管理セッションを終了する。
HELP	InfoServer コマンドのヘルプ・テキストを表示する。
INITIALIZE	読み書き可能ディスクを InfoServer ディスク用にフォーマットする。
LOOP	有効な任意の InfoServer のコマンドを自動的に繰り返させる。
MONITOR	有効な InfoServer のコマンドを 3 秒毎に繰り返させ、画面をクリアしてカーソルをホーム・ポジションに置く。
PURGE	VXT ソフトウェアの古いバージョンをパージする。
REBOOT	サーバをシャットダウンし、リブートする。

表 11-1 InfoServer コマンド (続き)

コマンド	機能
RECORD	InfoServer ディスクまたはパーティションのデータをコンパクト・ディスクに記録する。
RESTORE	サーバを以前に保存したシステム構成の状態にリセットする。
RETRIEVE	BACKUP コマンドにより保存した InfoServer 形式のディスクをリストアする。
REWIND	InfoServer テープを巻戻す。
SAVE	後でサーバをリポートするとき回復できるように、構成とサービスのデータを保存する。
SET	パーティション、サービス、またはサーバのパラメータを設定する。
SHOW	サーバのパラメータおよびカウンタを表示する。
UNBIND	VDK1 仮想デバイスを削除し、リモート・サービスへの接続を終了する。
UNLOAD	InfoServer テープを巻戻しアンロードする。
UPDATE	1 つまたは複数の新しいソフトウェア製品あるいは機能をインストールする。
VERIFY	INITIALIZE コマンドでフォーマットしたデバイスのオンディスク構造の妥当性を検査する。
ZERO	内部サーバ・カウンタを 0 に設定する。

InfoServer システムにはヘルプ機能があり、パラメータ、修飾子、使用法の例など、サーバの各コマンドに関する情報を表示することができます。InfoServer コマンドについての詳細は、『InfoServer System Operations Guide』を参照してください。

11.4 InfoServer Client for OpenVMS 機能について

InfoServer Client for OpenVMS により、OpenVMS オペレーティング・システムを実行しているクライアントは、InfoServer システムが LAN 上で提供している仮想デバイスにアクセスすることができます。ソフトウェア・コンポーネントには、次のものがあります。

- LASTport ドライバ
LASTport ドライバにより、クライアントは信頼性の高いデータ転送サービスを得ることができる。このドライバは、仮想デバイス・サービス用の効率的なトランスポートとして、データ・リンク・ドライバおよび LASTport/Disk ドライバと会話する。また、LASTport ドライバは、プリミティブ・データ・キュー登録サービスなどの他のアプリケーションをサポートすることもできる。
- LASTport/Disk クライアント・ドライバ
LASTport/Disk クライアント・ドライバは、システムへの標準のブロック・デバイス・インタフェースを提供する。OpenVMS ファイル・システムは、LASTport/Disk クライアントがローカル・ディスク・ドライバであるかのように LASTport/Disk クライアントと会話する。LASTport/Disk クライアント・ドライバは未処理インタフェースとバッファード・インタフェースの両方をサポートする。
- LASTport/Tape クライアント・ドライバ
LASTport/Tape クライアント・ドライバにより、OpenVMS クライアントは、InfoServer システムに接続されている SCSI テープにローカル・デバイスとしてアクセスし、これを使用することができる。
- LASTCP ユーティリティと LADCP ユーティリティ
この 2 つのユーティリティにより、使用しているシステムで InfoServer Client ソフトウェアを起動し、トランスポートの状態を監視し、InfoServer デバイス・サービスの構成と保守を行うことができる。11.5 項「LASTCP ユーティリティの機能について」と 11.6 項「LADCP ユーティリティの機能について」でこれらのユーティリティについて説明するが、詳細は『InfoServer Client for OpenVMS LASTCP and LADCP Utilities』を参照。

11.5 LASTCP ユーティリティの機能について

InfoServer Client for OpenVMS は、LASTport プロトコルを使用して拡張 LAN 上の InfoServer システムと通信します。このプロトコルは、OpenVMS デバイス・ドライバ ESS\$LASTDRIVER でインプリメントされます。

LASTport 制御プログラム (LASTCP) ユーティリティは、ESS\$LASTDRIVER の制御と診断を行うための管理インタフェースです。LASTCP を使用して次の作業を行うことができます。

- ESS\$LASTDRIVER の起動と停止
- サーキット、回線、ノード、および ESS\$LASTDRIVER のカウンタの表示
- ノード特性の表示
- 既知のクライアントおよびサーバの表示
- LASTport の状態の表示
- カウンタのリセット

LASTCP ユーティリティの説明では次の項目を取り上げます。

- LASTCP ユーティリティの起動と終了
- LASTCP のコマンドの要約
- InfoServer Client for OpenVMS の自動起動

11.5.1 LASTCP ユーティリティの起動と終了

LASTCP を使用するためには、特別な場合を除き、通常の特権が必要です。LASTCP を起動するには、次のコマンドを入力します。

```
$ RUN SYS$SYSTEM:ESS$LASTCP
%LASTCP-I-VERSION, ESS$LASTDRIVER V1.5 is running
LASTCP>
```

LASTCP コマンドは LASTCP> プロンプトに対して入力します。LASTCP ユーティリティを終了するには、LASTCP> プロンプトの後に EXIT を入力するか、または Ctrl/Z を押します。

次の例に示すように、DCL の文字列代入文を使用して、単一の LASTCP コマンドを実行することもできます。

```
$ LASTCP ::= $ESS$LASTCP
$ LASTCP SHOW CLIENTS
```

LASTCP は SHOW CLIENTS コマンドを実行してから、制御を DCL のコマンド・レベルに戻します。

11.5.2 LASTCP コマンドの要約

表 11-2 「LASTCP コマンド」に、LASTCP コマンドについてまとめます。

表 11-2 LASTCP コマンド

コマンド	機能
EXIT	ユーザを DCL のコマンド・レベルに戻す。
HELP	LASTCP コマンドのヘルプ・テキストを表示する。
SHOW CIRCUIT COUNTERS	サーキット・カウンタを表示する。
SHOW CLIENTS	既知のクライアントを表示する。
SHOW LINE COUNTERS	回線カウンタを表示する。
SHOW NODE CHARACTERISTICS	ノード特性を表示する。
SHOW NODE COUNTERS	ノード・カウンタを表示する。
SHOW SERVERS	既知のサーバを表示する。

表 11-2 LASTCP コマンド (続き)

コマンド	機能
SHOW STATUS	ローカルの状態を表示する。
SHOW TRANSPORT COUNTERS	トランスポート・カウンタを表示する。
START TRANSPORT	LASTDRIVER を起動する。
STOP TRANSPORT	LASTDRIVER を停止する。
ZERO COUNTERS	カウンタをリセットする。

一意に認識できれば、LASTCP コマンドを短縮することができます。たとえば、SHOW SERVERS コマンドは SH SE と指定することができます。

LASTCP にはヘルプ機能があり、各コマンドとそのパラメータ、修飾子に関する情報と使用法の例を表示することができます。LASTCP コマンドについての詳細は、『InfoServer Client for OpenVMS LASTCP and LADCP Utilities』を参照してください。

11.5.3 InfoServer Client for OpenVMS の自動起動

InfoServer Client for OpenVMS は、ESS\$STARTUP コマンド・プロシージャを使って起動する必要があります。システムのリブート時にこのソフトウェアが必ず自動起動するようにするには、SYSTARTUP_VMS.COM 内部からこのスタートアップ・プロシージャを実行します。

作業方法

1. SCSNODE (使用しているシステムのノード名パラメータ) の値を決定する。このパラメータが空文字列 (省略時の値) として定義されていると、InfoServer Client for OpenVMS は起動しない。

DECnet for OpenVMS を実行している場合、または実行を予定している場合、SCSNODE はシステムの DECnet ノード名として定義しなければならない。DECnet を実行しない場合で、かつ、システムが OpenVMS Cluster のメンバである場合、SCSNODE は SCS システム名 (クラスタ内で一意の 1 文字から 6 文字のノード名) として定義しなければならない。

SCSNODE の値を決定するには、次のコマンドを入力して SYSMAN を起動し、パラメータを表示する。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> PARAMETERS USE CURRENT
SYSMAN> PARAMETERS SHOW SCSNODE
```

2. SCSNODE が空文字列として定義されている場合は次の手順に従う。
 - a. 次の形式でコマンドを入力する。ここで、**ノード名**はシステムの DECnet ノード名、あるいは DECnet for OpenVMS を実行する予定のない場合は SCS システム名。
PARAMETERS SET SCSNODE "ノード名"

次に例を示す。

```
SYSMAN> PARAMETERS SET SCSNODE "MYNODE"
```

- b. 次のコマンドを入力して、新しい値をパラメータ・ファイルに書き込み、SYSMAN を終了する。

```
SYSMAN> PARAMETERS WRITE CURRENT
SYSMAN> EXIT
```

- c. 次の形式の行を AUTOGEN パラメータ・ファイル SYS\$SYSTEM:MODPARAMS.DAT に追加し、SCSNODE パラメータを定義する。
SCSNODE = "ノード名"

次に例を示す。

```
SCSNODE = "MYNODE"
```



注記: SCSNODE の最大サイズ 6 文字は、厳密に制限されます。このサイズとして 6 文字を超える値がシステム・パラメータ・ファイル内で設定されていた場合、SCSNODE の値は SYSBOOT によって短縮されます。

3. 任意のエディタを起動し、SYS\$MANAGER:SYSTARTUP_VMS.COM を編集して、InfoServer Client ソフトウェアを起動するコマンドを見つける。次に例を示す。

```
$ @SYS$STARTUP:ESS$STARTUP DISK
```

なお、CLIENT と DISK というパラメータは同じものである。コマンドの先頭に DCL コマンドの区切り文字 (!) がある場合はこれを削除する。テープ機能を使用できるようにしたい場合は、次のようにコマンド行に TAPE パラメータを追加する。

```
$ @SYS$STARTUP:ESS$STARTUP DISK TAPE
```

4. SYSTARTUP_VMS.COM により DECnet for OpenVMS のスタートアップ・プロシージャ (SYS\$MANAGER:STARTNET.COM) が起動される場合は、SYSTARTUP_VMS.COM が、STARTNET.COM を起動して **から**、InfoServer Client for OpenVMS のスタートアップ・プロシージャが起動されるようにする。

次に、ネットワークのスタートアップ・コマンド行の後に、InfoServer Client for OpenVMS のスタートアップ・コマンド行が続いている例を示す。なお、TAPE パラメータを指定しなかった場合は、ディスク機能しか起動されない。

```
$ @SYS$MANAGER:STARTNET
```

```
$ @SYS$STARTUP:ESS$STARTUP DISK TAPE
```

5. また、ファイル SYS\$STARTUP:ESS\$LAST_STARTUP.DAT を編集して、LASTport トランスポート用にスタートアップ修飾子を指定することもできる。『InfoServer Client for OpenVMS LASTCP and LADCP Utilities』を参照。

11.5.4 DECnet の起動または停止による InfoServer クライアントの起動失敗

DECnet を起動して停止した後、そのシステムで InfoServer クライアント ソフトウェアを起動使用とすると失敗します。ファイル SYS\$MANAGER:ESS\$STARTUP.LOG に、次のメッセージが出力されます。

```
%ESS-I-NONET ESS started before DECnet. 4-MAR-2000 16:36:39.29
```

このとき、InfoServer クライアントを起動する必要がある場合には、次のコマンドを実行すると、LAST 制御プログラムで LASTport トランスポートを起動できます。

```
$ MCR ESS$LASTCP  
LASTCP> START
```

このコマンドは、トランスポートを起動します。トランスポートが起動されると、InfoServer クライアントを起動できるようになります。

```
$ @SYS$STARTUP:ESS$STARTUP DISK
```

トランスポートがすでに起動されているため、InfoServer クライアントの起動が成功します。

11.5.5 構成済みであるが媒体に接続されていない複数コントローラ (Alpha および I64)

OpenVMS Alpha システムまたは I64 システムに、イーサネット・コントローラと FDDI コントローラが複数構成されている場合、次のどちらかの状況で InfoServer クライアント・トランスポート (LASTport) に問題が発生する可能性があります。

- ネットワーク・ケーブルに、イーサネット・コントローラと FDDI コントローラが接続されていない。
- FDDI コントローラはネットワーク・ケーブルに接続されているが、FDDI のリングが機能していない。たとえば、FDDI ハードウェアにの電源がオフになっている、または壊れている場合など。

ネットワークで利用できるすべてのサービスにアクセスできなくなったり、4 つ以上のコントローラが構成されている場合にはシステムがクラッシュしたりするなどの問題が発生します。これらの問題を回避するためには、媒体に接続されているコントローラだけを指定します。最初に SYS\$STARTUP:ESS\$LAST_STARTUP.DAT データ・ファイルを編集して、接続されているコントローラだけを指定してから、システムを再起動することをお勧めします。

特定のコントローラ構成では、接続されていないコントローラを指定すると、次のコマンド・シーケンスを実行した場合にシステムがクラッシュする可能性があります。

```
$ MCR ESS$LASTTCP
LASTTCP> STOP
```

次の例では、SYS\$STARTUP:ESS\$LAST_STARTUP.DAT ファイルの編集方法を説明します。最初に編集前のファイルを示し、次に編集後のファイルを示します。

```
!++
! This file will be used to set the appropriate LASTTCP qualifiers. The following
! LASTTCP qualifiers: ALL_CONTROLLERS, CHECKSUM, TRANSMIT_QUOTA, or SLOW_MODE
! can be set by using the following statement format:
! LASTTCP qualifier = 1 to enable e.g. SLOW_MODE = 1 enables SLOW_MODE
! LASTTCP qualifier = 0 to disable e.g. SLOW_MODE = 0 disables SLOW_MODE
! The remaining LASTTCP qualifiers will require the appropriate value settings.
! DEVICE = (list-of-devices)
! TIMEOUT = n minimum interval in seconds
! CIRCUIT_MAXIMUM = n maximum number of nodes
! GROUP = n Group number
! NODE_NAME = name Node name
! CONTROLLERS = ([{controller letter,}...]) Controller list
! TRANSMIT_QUOTA = n Number of transmit buffers
!--
ALL_CONTROLLERS = ON
```

次に、編集済みの SYS\$STARTUP:ESS\$LAST_STARTUP.DAT ファイルを示します。この例では、システムに ESA, ETA, EXA, EZA の各コントローラが構成され、ESA コントローラだけがイーサネット・ケーブルに接続されている場合を想定しています。

```
!++
! This file will be used to set the appropriate LASTTCP qualifiers. The following
! LASTTCP qualifiers: ALL_CONTROLLERS, CHECKSUM, TRANSMIT_QUOTA, or SLOW_MODE
! can be set by using the following statement format:
! LASTTCP qualifier = 1 to enable e.g. SLOW_MODE = 1 enables SLOW_MODE
! LASTTCP qualifier = 0 to disable e.g. SLOW_MODE = 0 disables SLOW_MODE
! The remaining LASTTCP qualifiers will require the appropriate value settings.
! DEVICE = (list-of-devices)
! TIMEOUT = n minimum interval in seconds
! CIRCUIT_MAXIMUM = n maximum number of nodes
! GROUP = n Group number
! NODE_NAME = name Node name
! CONTROLLERS = ([{controller letter,}...]) Controller list
! TRANSMIT_QUOTA = n Number of transmit buffers
!--
```



注意:

ESS\$LAST_STARTUP.DAT ファイルは、省略時には SYS\$COMMON:[SYS\$STARTUP] に格納されています。その他のシステム・ルートに影響しないように、編集済みファイルを SYS\$SPECIFIC:[SYS\$STARTUP] に置くことができます。

11.5.6 スタートアップの制限事項: PATHWORKS と RSM

PATHWORKS またはリモート・システム・マネージャ (RSM)、あるいはこの両方がインストールされている場合、InfoServer Client for OpenVMS のスタートアップを実行してからでないと、PATHWORKS または RSM、あるいはこの両方のスタートアップを行うことはできません。

```
$ @SYS$MANAGER:STARTNET
```

```
$ @SYS$STARTUP:ESS$STARTUP DISK TAPE
```

```
$ @SYS$STARTUP:PCFS_STARTUP
```

```
$ @SYS$STARTUP:RSM$SERVER_STARTUP
```

InfoServer Client for OpenVMS には、PATHWORKS と RSM の両製品で共用するデバイス・ドライバと制御プログラムがあります。InfoServer Client for OpenVMS の全コンポーネントの先頭には、ESS\$ という接頭辞が付いています。InfoServer Client for OpenVMS のドライバと制御プログラムは、InfoServer Client サポートの他に、PATHWORKS と RSM の両方のために必要なすべてのサポートを提供します。InfoServer Client for OpenVMS のスタートアップは、PATHWORKS あるいは RSM のスタートアップ・プロシージャを実行する前に、サイト別スタートアップで実行する必要があります。

11.5.7 スタートアップの制限事項: SYSMAN

サブプロセスから InfoServer Client for OpenVMS を起動することはできません。これは、OpenVMS のシステム管理ユーティリティ (SYSMAN) は、サブプロセスを使用して、遠隔ノード上のタスクを完了するので、SYS\$STARTUP:ESS\$STARTUP プロシージャを実行するために SYSMAN を使用することはできないためです。

11.5.8 ユーザ・アカウントの必要条件

InfoServer Client for OpenVMS を使用して作業する場合、使用しているシステムのユーザ・アカウントには、次の特権とクォータが必要です。

- LADCP BIND コマンドに /GROUP 修飾子を指定する場合は GRPNAM 特権、LADCP BIND コマンドに /SYSTEM 修飾子を指定する場合は SYSNAM 特権が必要。
- 少なくとも、省略時の UAF アカウント・クォータが必要。

アカウントの特権およびクォータのチェックと変更の方法については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の AUTHORIZE の節を参照してください。

11.5.9 システム・パラメータ MAXBUF の必要条件

LASTCP ユーティリティの SHOW の全機能を使用する場合は、システム・パラメータ MAXBUF の値を 32,000 以上に設定する必要があります。

11.6 LADCP ユーティリティの機能について

OpenVMS システムの LASTport/Disk プロトコルおよび LASTport/Tape プロトコルの構成と制御は、LAD 制御プログラム (LADCP) を使用して行います。LASTport/Disk サービスと

LASTport/Tape サービスを使用する OpenVMS システムを、クライアント・システムといいます。LADCP を使用して、次の作業を行うことができます。

- サービスへの **バインド** の確立。バインドにより、ローカルの OpenVMS システム上に新しい DADn: 仮想ディスク・ユニット、または MADn: 仮想テープ・ユニットが作成される。
- サービスへのバインドの削除。

サービス・アクセス・パスワードを使って、サービス・アクセスを制御することができます。また、サービスの書き込み保護も行うことができます。この場合、DADn: あるいは MADn: デバイス・ユニットのローカルの OpenVMS ユーザがそのユニットに書き込み操作を行おうとすると、エラー・メッセージが表示されます。

これらのプロトコルにより、InfoServer システム上の記憶デバイスが、OpenVMS システムにローカルで接続されているかのようにアクセスすることができます。このように、複数の OpenVMS クライアント・システムで 1 つの読み込み専用媒体を共用することができるので、同じデバイスと媒体を用意する必要がなくなります。

DADn: および MADn: デバイス・ユニットを、**仮想デバイス・ユニット**ともいいます。これらは、遠隔サーバ上のボリュームに対して、ローカルの OpenVMS のコンテキストを表現します。DADn: ユニットの制御する OpenVMS ドライバを ESS\$DADDRIVER, MADn: ユニットの制御する OpenVMS ドライバを ESS\$MADDRIVER といいます。

LASTport/Disk プロトコルと LASTport/Tape プロトコルは、LASTport トランスポートに依存しています。SYS\$STARTUP 内の ESS\$STARTUP.COM コマンド・プロシージャは、LASTport トランスポート・ドライバ ESS\$LASTDRIVER, および ESS\$DADDRIVER と ESS\$MADDRIVER を自動的にロードします。



注意:

サイト別スタートアップ・コマンド・プロシージャには、ESS\$STARTUP.COM への呼び出しが必要です。DECnet を使用している場合は、その呼び出しを DECnet を起動する SYS\$MANAGER:STARTNET.COM コマンドの **後に**置く必要があります。11.5.3 項「InfoServer Client for OpenVMS の自動起動」を参照してください。

11.6.1 LADCP ユーティリティの起動と終了

LADCP を起動するには、次のコマンドを使用します。

```
$ RUN SYS$SYSTEM:ESS$LADCP
LADCP>
```

LADCP コマンドは、LADCP> プロンプトに入力することができます。

また、次の例に示すように、DCL の文字列代入文を使用することにより、単一の LADCP コマンドを実行することもできます。

```
$ LADCP ::= $ESS$LADCP
$ LADCP BIND CD_DOC_00661 /NOWRITE
```

LADCP は、BIND コマンドを実行してから DCL のコマンド・レベルに制御を戻します。

LADCP を終了するには、LADCP> プロンプトの後に EXIT を入力するか、あるいは Ctrl/Z を押します。

11.6.2 LADCP コマンドの要約

表 11-3 「LADCP コマンド」に、LADCP コマンドについてまとめます。

表 11-3 LADCP コマンド

コマンド	機能
BIND	サービスのバインドを確立し、デバイス・ユニットを作成する。
DEALLOCATE	ユニット制御ブロック (UCB) を削除せずにサービスへのアクティブな接続を終了する。
EXIT	ユーザを DCL のコマンド・レベルに戻す。
HELP	LADCP コマンドのヘルプ・テキストを表示する。
SHOW SERVICES	LAN 上で使用できる InfoServer システムが提供するサービスを表示する。
UNBIND	確立した LAD サービスのバインドを終了する。

LADCP にはヘルプ機能があり、LADCP の各コマンドのパラメータ、修飾子、および使用法の例に関する情報を表示することができます。LADCP コマンドについての詳細は、『InfoServer Client for OpenVMS LASTCP and LADCP Utilities』を参照してください。

11.6.3 InfoServer デバイスを自動的に使用可能にする方法

一連の LADCP の BIND コマンドを SYSTARTUP_VMS.COM に追加すると、システムがブートするたびに、遠隔 InfoServer デバイスを使用可能にするよう設定できます。BIND コマンドについての詳細は、『InfoServer Client for OpenVMS LASTCP and LADCP Utilities』を参照してください。

作業方法

1. SYSTARTUP_VMS.COM を編集し、InfoServer Client ソフトウェアを起動するコマンドを見つけたら、次に例を示す。

```
@SYS$STARTUP:ESS$STARTUP DISK TAPE
```

このコマンドにより、ディスク機能とテープ機能を持つソフトウェアが起動される。

2. 次のコマンドを追加して、LADCP を起動する。

```
$ RUN SYS$SYSTEM:ESS$LADCP
```

3. このコマンドの直後に、次の形式で BIND コマンドを追加し、InfoServer の任意のコンパクト・ディスクあるいはハード・ディスクを仮想デバイス・ユニットとして使用できるようにする。

```
BIND [/修飾子,...] サービス名
```

テープ・デバイスを使用できるようにするためには、その他の修飾子に加えて、/TAPE 修飾子も指定する必要がある。

```
BIND/TAPE [/修飾子,...] サービス名
```

サービス名には、InfoServer のデバイス・サービスの名前を指定する。通常、サービス名は InfoServer システムによるアクセス先のボリュームのラベルである。BIND コマンドについての詳細は、『InfoServer Client for OpenVMS LASTCP and LADCP Utilities』を参照。

4. EXIT コマンドを追加して、LADCP を終了する。
5. MOUNT コマンドを次の形式で追加し、作成した仮想デバイス・ユニットを公用デバイスとして使用できるようにする。

```
MOUNT/SYSTEM/NOASSIST デバイス名 ボリューム・ラベル
```

デバイス名には、デバイスの名前、**ボリューム・ラベル**には、デバイスに割り当てるボリューム・ラベルを指定する。MOUNT コマンドについての詳細は、『OpenVMS DCL ディクショナリ』の MOUNT の節を参照。

例

次のコマンドを SYSTARTUP_VMS.COM で実行すると、InfoServer Client ソフトウェアが起動され、InfoServer デバイス DAD\$OPENVMSV72 が使用可能になります。

```
$ @SYS$STARTUP:ESS$STARTUP DISK
$ RUN SYS$SYSTEM:ESS$LADCP
  BIND OPENVMSV72
  EXIT
$ MOUNT/SYSTEM/NOASSIST DAD$VMS055 VMS055
```

この例では、InfoServer システムに接続されているコンパクト・ディスク・ドライブにロードされた、OpenVMS バージョン 7.2 の統合ディストリビューション (CONdisk) コンパクト・ディスクが、サーバ上で仮想デバイス・ユニットとして使用可能になり、公用デバイスとしてマウントされます。

第12章 LAT ソフトウェアの管理

この章では、LAT ソフトウェアの機能と、システム上に LAT ソフトウェアをインプリメントして、管理するために必要な作業について説明します。

この章の内容

この章では次の作業について説明します。

作業	参照箇所
LAT プロトコルのスタートアップ	12.5 項 「LAT プロトコルのスタートアップ」
LAT 特性のカスタマイズ	12.6 項 「LAT 特性のカスタマイズ」
サービスの定義	12.6.1 項 「付加サービスの定義」
ポートの設定	12.6.2 項 「ポートの設定」
プリンタの設定	12.6.2.1 項 「プリンタの設定」
特殊アプリケーション・サービスの設定	12.6.2.2 項 「特殊アプリケーション・サービスの設定」
キュー登録された外部からの接続要求の許可	12.6.3 項 「外部からの接続要求のキュー登録」
外部への LAT 接続の許可	12.2.2.2 項 「接続の設定」
LATACP データベースのサイズの管理	12.7 項 「LATACP データベースのサイズの管理」

さらに、次の項目について説明します。

項目	参照箇所
LAT プロトコルについて	12.1 項 「LAT プロトコルについて」
LAT ネットワークについて	12.2 項 「LAT ネットワークについて」
LAT 構成について	12.3 項 「LAT 構成について」
LAT 制御プログラム・ユーティリティについて	12.4 項 「LATCP ユーティリティについて」

12.1 LAT プロトコルについて

オペレーティング・システムは、ローカル・エリア・ネットワーク (LAN) 内で**ターミナル・サーバ**および他のシステムと通信を行うときに LAT (ローカル・エリア・トランスポート) ソフトウェアを使用します。ターミナル・サーバは、ターミナル、モデム、またはプリンタの LAN への接続を専門に行う通信デバイスです。ターミナル・サーバには次の特長があります。

- 安価な方法で多くのユーザ・ターミナルをコンピュータに接続することができる。
- 必要なケーブル長が短くてすむ。
- コンピュータにアクセスできるデバイスの数を最大にすることができる。

LAT **プロトコル**をインプリメントする LAT ソフトウェアにより、オペレーティング・システムはターミナル・サーバがアクセスすることができる資源、すなわち**サービス**を提供することができます。LAT サービスを提供するシステムを**サービス・ノード**と呼びます。また、ノードは、LATCP を使って外部への接続を開始し、DCL の SET HOST/LAT コマンドを使用することによって LAT サービスにアクセスすることができます。この章の以降の説明では、「サーバ」という用語は、専用のターミナル・サーバと他の LAT サービスへの外部アクセスを可能にするノードの両方の意味で使用しています。

LAT サービスは、コンピュータ・システムのすべての資源で構成することも、アプリケーション・プログラムのような特定の資源だけで構成することもできます。システムを**汎用タイムシェアリング・サービス**として設定することにより、システムのすべての資源を LAN 上の各

ユーザから使用可能にしたり、システム上の特定のサービス (アプリケーション・プログラム) へのアクセスを制限したりできます。この章および『OpenVMS I/O User's Reference Manual』では、専用のアプリケーション・プログラムへのアクセスを設定するための手順の概要を説明します。

12.1.1 LAT プロトコルの機能

LAT プロトコルは、ターミナル・サーバおよびコンピュータがイーサネットまたは FDDI (Fiber Distributed Data Interconnect) などを利用した LAN 上で相互に通信を行うことを可能にします。LAT プロトコルは、ターミナルなどのデバイスと LAN のシステム資源 (サービス) とを照合します。LAT ターミナルはアクセスしようとするコンピュータ (サービス・ノード) に直接接続されていないため、ターミナルから出されるサービス要求を受け付け、要求を出したターミナルと要求されたサービスを提供するコンピュータとを照合する機能がローカル・サーバに必要となります。

LAT プロトコルを使用することにより、オペレーティング・システムは提供可能なサービスを LAN 上に通知します。サーバは、LAN の通知をリッスンして、サービス情報のデータベースを構築します。このデータベースにより、サーバはユーザ・ターミナルからシステム・サービスを要求されたときに、そのサービスを提供するシステムを特定することができます。たとえば、あるユーザ・ターミナルからオペレーティング・システム上の汎用処理サービスまたはデータ・エントリ・プログラムを要求された場合、サーバは LAT プロトコルを使用して要求側のターミナルとオペレーティング・システムとの間の接続を確立して、保持します。

オペレーティング・システムは、ターミナル・サーバに対してサービスを要求できる場合があります。LAT プロトコルを使用すると、システムは、ターミナル・サーバに直接接続されているプリンタなどのデバイスへの接続を要求することができます。

12.1.2 LAT プログラムの利点

システム上で LAT プロトコルを使用することには、次のような多くの利点があります。

- LAT プロトコルにより、ローカル・エリア・ネットワーク上の任意のコンピュータの資源をそのネットワーク上のすべてのユーザから利用可能にすることができる。
- 汎用の処理資源に加え、ターミナル、プリンタ、モデムをローカル・エリア・ネットワーク上の複数のシステムから利用できるように設定することができるため、これらの資源を効果的に使用することができる。また、ネットワーク上のシステムの 1 つをシャットダウンしなければならない場合でも、資源を利用可能なままにしておくことができる。
- データ・エントリ・プログラムやニュース・サービスのようなアプリケーション・プログラムも資源として設定することができる。ユーザが資源への接続を要求した場合、LAT プロトコルはそのアプリケーション・プログラムへの直接の接続を設定するため、ユーザは新たなログイン手続きを行う必要がない。
- LAT プロトコルは、負荷分散機能と回復メカニズムを備えているため、ユーザは最も安定した信頼性の高いサービスを受けることができる。システムは、各サービスが現在どのくらい利用できるかをブロードキャスト・メッセージの中で示すため、サーバは最も使用されていないノードを接続先として選択することができる。あるノードが何らかの理由でサービスを提供できなくなった場合、サーバは代替サービスへの接続を試みる。
- ユーザはターミナル上で複数のコンピューティング・セッションを確立することができ、いくつかの異なるコンピュータに接続し、あるコンピューティング・セッションから別のセッションに容易に切り替えることができる。また、1 つのセッションの作業を中断して別のセッションに移り、後で元のセッションに戻ってそのセッションの作業を再開することができる。したがって、ファイルまたはアカウントをいったんクローズして再びオープンしたり、以前のセッションの同じポイントに戻ったりする手間が省ける。
- 結果としてシステム性能が向上する。サーバはメッセージを単一の LAN インタフェースにまとめて送るため、ネットワーク・トラフィックが少なくなり、ターミナル、モデム、プリンタがコンピュータに物理的に接続されているシステムで行われるコンピュータ割り込みの数は減少する。

12.2 LAT ネットワークについて

ターミナル・サーバおよびオペレーティング・システムが LAT プロトコルを使用しているローカル・エリア・ネットワークのことを **LAT ネットワーク** と呼びます。LAT ネットワークは、同じ LAN 上で他のプロトコルと共存することができます。LAT プロトコルはターミナル・サーバおよびオペレーティング・システムの両方で動作し、LAN 上でデータが安全に送信される設計になっています。

LAT ネットワークのコンポーネントは次のとおりです。

コンポーネント	参照箇所
サービス・ノード	12.2.1 項「サービス・ノード」
ターミナル・サーバ・ノード	12.2.2 項「ターミナル・サーバ・ノード」
外部への接続が可能なノード	12.2.3 項「外部への接続が可能なノード」
LAN ケーブル	12.2.4 項「LAT ネットワークのコンポーネント」

サービス・ノードはローカル・ネットワークのためにシステム資源を提供します。一方、ターミナル・サーバ・ノード(または外部への接続が可能なノード)は、ユーザ・ターミナルまたはアプリケーション・プログラムから要求があると、ターミナル、モデム、またはプリンタをシステム資源に移植します。

LAT ネットワークでは、サービスに**アクセス**するノードを**マスタ・ノード**と呼ぶことがよくあるので注意してください。この呼び方によって、このノードをサービスを**提供する**だけのノードと区別できます。

LAT 制御プログラム (LATCP) を使用して、システムに合わせて LAT 特性を構成することができます。LATCP を使用すると、システムのサポートする内容を次のように設定することができます。

- 外部からのアクセスのみ
- 外部へのアクセスのみ
- 内部および外部への両方向のアクセス

外部から内部への LAT 接続をサポートするシステムは、**サービス・ノード**です。LATCP を使用して、外部からのアクセスと外部へのアクセスをともにサポートしないようにシステムを設定することもできます。

12.2.1 サービス・ノード

サービス・ノードは、LAT ネットワークを構成するノードの種類の一つです。LAT ネットワーク上では、OpenVMS オペレーティング・システムが稼働していないノードも OpenVMS ノードとともに使用することができます。サービス・ノードは、ユーザおよびデバイスに資源を提供する LAN 上の個々のコンピュータです。OpenVMS オペレーティング・システムには LAT プロトコルが含まれているため、どの OpenVMS システムでも LAT ネットワーク上でサービス・ノードとして構成することができます。

12.2.1.1 サービスの種類

各ノードはその資源を**サービス**として提供します。多くの場合、各ノードからは汎用処理サービスを提供しますが、制限されたサービスや特殊なアプリケーション・サービスを提供することもできます。また、1つのノードから提供するサービスの一部あるいはすべてを特殊なアプリケーションにすることが可能です。

たとえば、サービス・ノードから提供できるサービスには次のものがあります。

- 汎用処理
- データ・エントリ
- 株式相場

汎用処理サービスでは、汎用のシステム環境を使用することができます。一方、データ・エントリや株式相場のサービスの場合はその環境が制限され、サービス・ノードのアプリケーション・サービスへの接続は行われますが、他の部分への接続は行われません。

各サービスはシステム管理者によって割り当てられた名前によって区別されます。OpenVMS Cluster では、サービス名はクラスタ名と同じにしてください。独立したノードでは、サービス名はそのノード名と同じにしてください。特殊なサービス・アプリケーションを使用するサービスには、そのアプリケーションの名前を付けるようにします。

12.2.1.2 サービスの通知

サービス・ノードはそのサービスの内容を一定の間隔で LAN 上に通知します。ターミナル・サーバ(および外部への接続が可能な OpenVMS システム)は、それによって各ネットワーク・サービスが利用可能かどうかを知ることができます。通知される内容は、物理ノード名、サービス名、サービスの内容説明、そのサービスの現在の利用量です。サーバは LAN への通知をリッスンして、情報をデータベースに記録します。外部への接続が可能なノードでは、このデータベースは LAT 補助制御プロセス (LATACP) によって保守されます。LATACP データベースの管理についての詳細は、12.7 項「LATACP データベースのサイズの管理」を参照してください。

ユーザ・ターミナルまたはアプリケーション・プログラムからサービスの要求を受け取るたびに、サーバ・ノードは適切なサービス・ノードに接続を行います。

LATCP コマンド SET NODE に/NOANNOUNCEMENTS 修飾子を指定して使用することにより、マルチキャスト・サービスからローカル・ノードを無効にすることができます。ただし、遠隔ノードは、ローカル・ノードに接続するためには、LAT プロトコル・バージョン 5.2 (またはそれ以上) の LAT サービス応答機能に依存しなければならないため、新しいモデルのターミナル・サーバおよびホストがあるネットワーク環境 (LAT ホスト、ターミナル・サーバ、および PC がすべて、LAT プロトコルのバージョン 5.2 以上を実行している) でのみ、この修飾子を使用してください。この環境以外で使用した場合、バージョン 5.2 より前の LAT プロトコルを実行しているシステム (DECserver 100, 200, 500 システムなど) は、LAT サービス通知を無効にしているシステムとは接続できません。

12.2.1.3 プリント要求

サービス・ノードがターミナル・サーバにサービスを要求できる場合がいくつかあります。最も一般的なケースは、システムがターミナル・サーバのポートに接続されているプリンタを使用する場合です。システムはプリント要求をターミナル・サーバのプリント・キューに登録します(このプリント・キューは OpenVMS のスタートアップ・プロシージャの中で設定されて、初期化されます)。キューに要求が登録されると、LAT シンビオント (大容量記憶デバイスとの間でデータを転送するプロセス) により、LAT ポート・ドライバに対して遠隔プリンタへの接続の確立と終了が要求されます。

LAT ポートに接続されているプリンタのキューを設定する方法については、『OpenVMS システム管理者マニュアル (上巻)』を参照してください。

12.2.2 ターミナル・サーバ・ノード

ターミナル・サーバ・ノードは、LAT ネットワークを構成するノードのもう 1 つの種類です。通常、ターミナル・サーバ・ノードはそれがサポートするターミナルおよびプリンタの近くに存在します。ターミナルおよびプリンタは、ケーブルによってターミナル・サーバに物理的に接続され、LAN ケーブルには直接接続されません。LAN ケーブルはターミナル・サーバに物理的に接続されます。

12.2.2.1 サービス・ノードの位置の特定

ターミナル・サーバはネットワーク上の各ノードからの通知に基づいてサービスのディレクトリを作成し、それを管理します。ターミナル・サーバはターミナル・ユーザから要求を受け取ると、サービス・データベースを検索し、要求されたサービスを提供するコンピュータの位置を特定します。

ターミナル・サーバは要求されたサービスを提供するノードを探すだけでなく、そのノードのサービスの利用量を調べることがあります。要求されたサービスが複数のノードから提供されている場合、サービスの利用量が最も少ないノードを選択し、そのノードと要求側のユーザ・ターミナルとの間に論理接続を確立します。

12.2.2.2 接続の設定

1つの論理接続により、1つのターミナル・サーバ・ノードからサービス・ノードに向けられたすべてのデータが運ばれます。つまり、サーバは同じノードと通信を行っているすべてのターミナルからのデータを1つの接続上に結合します。ターミナル・サーバは、論理接続がまだ存在しない場合にだけサービス・ノードとの論理接続を確立します。

何らかの理由で接続に失敗すると、ターミナル・サーバは同じサービスを提供している別のノードを探し、そのノードとの接続を確立することによって、ユーザがコンピューティング・セッションを続行できるようにします。

ターミナルからの接続は1つにまとめられても、個々のターミナルはそのターミナル名によって区別されます。ターミナル名は2つの部分から構成されます。最初の部分はターミナル回線が接続されているターミナル・サーバ上のポートの名前を示し、2番目の部分はターミナル・サーバ・ノードの名前を示します。

12.2.2.3 サービス提供側のノード

通常、LAT ネットワークにおける要求側ノードはターミナル・サーバですが、サービス・ノードがターミナル・サーバに対してサービスを要求することがあります。最も一般的なケースとして、サービス・ノードがターミナル・サーバに接続されている遠隔プリンタにプリント要求をキュー登録することが挙げられます。

12.2.3 外部への接続が可能なノード

ノードは、外部からの接続だけ、外部への接続だけ、またはその両方が可能になるように設定することができます。ターミナル・サーバなどのノード(外部からの接続だけが可能なノードは除く)は、サービス・ノードの位置を特定して、接続を設定することができます。利用可能なノードとサービスに関する情報格納したデータベースは、LAT 補助制御プロセス (LATACP) によって保守されます。LATACP データベースの管理についての詳細は、12.7 項「LATACP データベースのサイズの管理」を参照してください。

外部への LAT 接続が可能になるように設定されたノード上では、ユーザは SET HOST/LAT コマンドを入力することによって LAT ネットワーク上の他のノードに接続することができます。詳細は、『OpenVMS DCL ディクショナリ』の SET HOST/LAT コマンドの定義を参照してください。

12.2.4 LAT ネットワークのコンポーネント

図 12-1 「LAT ネットワークの構成例」は、LAT ネットワークのコンポーネントを示しています。ネットワークは、サービス・ノード、ターミナル・サーバ・ノード、およびそれらを接続するイーサネット・ケーブルから構成されます。

図 12-1 「LAT ネットワークの構成例」にある NOE、LARRY、ALEXIS は、ネットワーク上のターミナル・サーバ・ノードにそれぞれのサービスを提供するノードです。

サービス・ノードのうちの NOE と LARRY は、コンピュータ・インターコネクト (CI) とスター・カプラによってクラスタ OFFICE を形成しています。クラスタ化されているこの2つのノードのサービス名はクラスタ名と同じになっています。この2つのサービス・ノードはともに OFFICE サービスを提供するため、ターミナル・サーバ・ノードは両方の OFFICE ノード上の作業負荷を調べ、利用量が少ない方のサービスを提供するノードに対して接続を確立します。

もう1つのサービス・ノード ALEXIS は LAT ネットワーク上の独立ノードのため、サービス名はノード名と同じになっています。

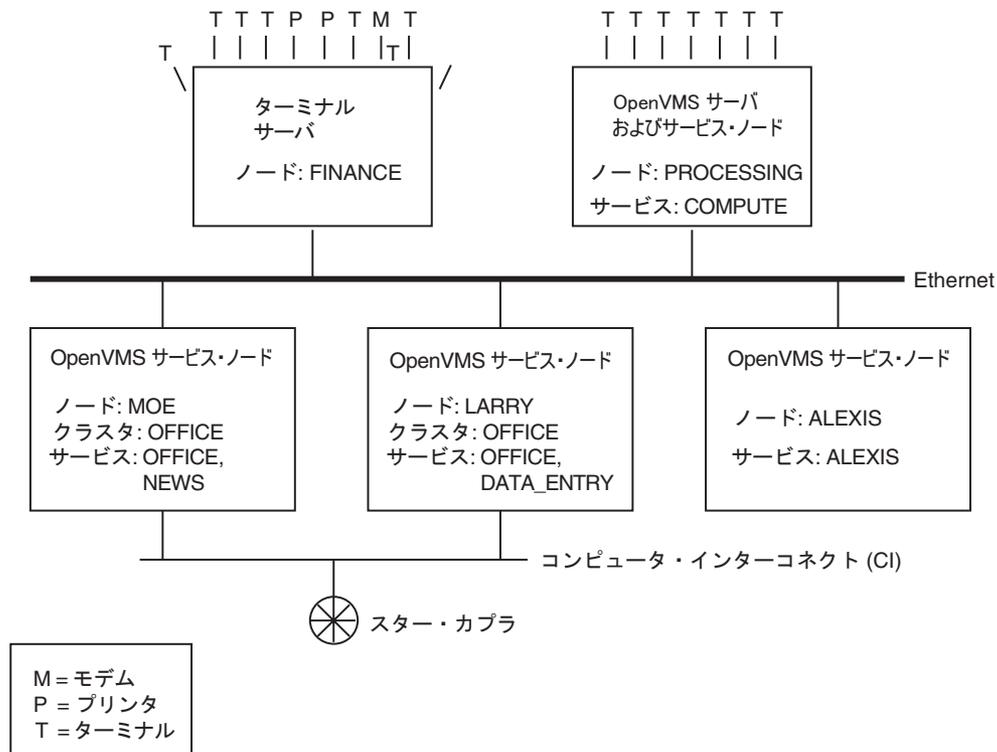
ノード NOE は一次的な OFFICE サービスに加えて NEWS と呼ばれるアプリケーション・サービスを提供します。この特殊化されたサービスを利用すれば、ユーザ・ターミナルはそのサー

ビス・ノードにログインしたり、そのノードの汎用コンピュータ資源に通常のアクセスをしなくても、オンライン・ニュース・サービスに直接接続を行うことができます。

図 12-1 「LAT ネットワークの構成例」のノード FINANCE はターミナル・サーバ・ノードです。このノードは多数の会話型ターミナル、モデム、およびプリンタをサポートします。ノード PROCESSING は、外部への接続が可能なノードです。このノードはいくつかの会話型ターミナルをサポートします。ノード FINANCE は 3 つのサービス・ノードのいずれからもプリント要求を受け付けることができます。ただし、各サービス・ノードでターミナル・サーバ上の遠隔プリンタをサポートするようにプリント・キューが設定されていることが必要です。

ノード PROCESSING はサービス・ノードとしても機能し、サービス COMPUTE を提供します。

図 12-1 LAT ネットワークの構成例



ZK1110AGE

12.3 LAT 構成について

LAT システムを構成する際には、システムが効率的に機能するように構成するために、LAT ソフトウェアとネットワークの関係を十分に理解しておく必要があります。以降の各項では、次のことを理解する上で役立つ情報を提供します。

- LAT ソフトウェアと OpenVMS Cluster ソフトウェアの関係
- LAT ソフトウェアと DECnet ソフトウェアの関係
- 複数の LAN アダプタを使用するネットワーク環境での LAN ソフトウェアの動作
- 大容量のバッファを使用できないイーサネット / FDDI 構成での LAT ソフトウェアの使い方

12.3.1 OpenVMS Cluster および DECnet と LAT との関係

LAT プロトコルは OpenVMS Cluster ソフトウェアとは独立して機能しますが、サービス・ノードは OpenVMS Cluster の概念を補うように構成します。そのために、OpenVMS Cluster 内の各ノード上にサービスを用意し、そのサービスにクラスタ名を割り当てます。ターミナル・サーバはクラスタ・サービスの可用性を調べて、利用量が最も少ないノードに対して接続を確立します。このように、LAT プロトコルはクラスタの負荷を分散する役割を果たします。ク

クラスタ内のノードの1つに障害が発生した場合、ターミナル・サーバは障害があった接続をクラスタ内の別のサービス・ノードに転送することができます。

LAT ソフトウェアはメッセージ伝送ファシリティとして DECnet を使用せず、代わりに独自の仮想サーキット層を使用して伝送メカニズムをインプリメントします。LAT と DECnet ソフトウェアは共通の LAN 環境でそれぞれから独立して機能します。サービス・ノードを DECnet ノードとしても使用する場合は、互換性のためにサービス・ノード名は DECnet ノード名と同じにしてください。

12.3.1.1 同じコントローラ上で実行する LAT と DECnet

イーサネット・ポートが DECnet と LAT の両方を実行している場合には、LAT ソフトウェアを起動する **前に** DECnet ソフトウェアを起動しなければなりません。最初に DECnet ソフトウェアを起動しなければ、存在するすべての LAN 接続が終了し、LAT を介したシステムへの再接続が不可能になることがあります。

12.3.1.2 異なるコントローラ上で実行する LAT と DECnet

DECnet がシステム上に構成されている場合 (あるいはシステムがクラスタに組み込まれている場合)、SCSSYSTEMID システム・パラメータはゼロ以外の値を含むことがあります。同じ論理 LAN に接続された LAN コントローラがシステムに複数台ある場合を除いて、通常これは問題になりません。

たとえば、システムに FDDI コントローラとイーサネット・コントローラがある場合には、FDDI コントローラに接続された FDDI リングとイーサネット・コントローラに接続されたイーサネット・セグメントが、10/100 LAN ブリッジ (FDDI からイーサネット) でブリッジされるように構成することができます。

このような構成では、SCSSYSTEMID が 0 でない場合、同じコントローラ上で LAT と DECnet とを実行 **しなければなりません**。同じコントローラ上で実行しなければ、DECnet が最初に起動するため、他のコントローラで起動する LAN が異常終了します。これは、LAT が起動時に AA-00-04-00-xx-xx アドレス (DECnet LAN アドレス) を使用しようとしても、DECnet がすでに別のコントローラでこのアドレスを使用しているため、データ・リンク層が、そのアドレスを使用した LAT の起動を妨げるためです (単一の論理 LAN においては、すべてのデータ・リンク・アドレスは固有のものでなければなりません。両方のコントローラが同じアドレスの使用を試みるため、固有ではなくなります)。

次のコマンドで LAT リンクを作成しても、LAN ドライバが SCSSYSTEMID に基づいたアドレスを使用しようとするため、失敗します。

```
LATCP> CREATE LINK LAT$LINK_2 /NODECNET
```

SCSSYSTEMID を 0 に設定すると、異なるコントローラ上で LAT と DECnet を構成することが可能になります。ただし、クラスタ環境では SCSSYSTEMID を 0 に設定することはできません。

12.3.2 複数の LAN アダプタの使用

1 つの LAT ノードに複数の LAN アドレスを使用すれば、同じ論理 LAN に接続された複数の LAN アダプタを持つシステムを構成できます。LAT ソフトウェアは各アダプタで同時に実行でき、接続も良好に維持されます。たとえば、仮想サーキットで 1 次パスを選び、すべての LAT メッセージ転送に使用した場合には、もとのパスが遮断されても、LAT ソフトウェアは、別のアダプタまたは論理パスを介して通信を継続できます。



注意:

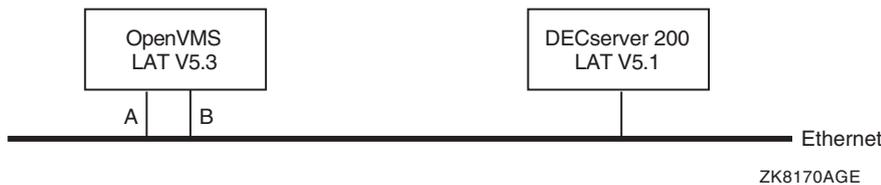
LAT プロトコルのバージョン 5.3 (バージョン 7.0 以降の OpenVMS オペレーティング・システムに含まれる) より前の LAT ソフトウェアのバージョンを実行しているノードの動作は少し異なります。したがって、システム構成に、バージョン 5.1 やバージョン 5.2 などの LAT ソフトウェアの初期のバージョンが含まれている場合には、この章で説明している相違点や考慮事項について注意してください。

12.3.2.1 サポートされる構成

複数の LAN アダプタで LAT を実行することはできますが、1 つの論理 LAN から別の論理 LAN に LAT を経路指定することはできません。次に示すのは、LAT プロトコル・バージョン 5.3 を実行しているノード (バージョン 5.2 および 5.1 を実行しているノードも含む) のためのサポートされている LAT 構成の例です。

この広く使用されている構成では、LAT バージョン 5.3 ソフトウェアを実行している OpenVMS システムが、同じ物理 LAN に接続されている 2 つのイーサネット・アダプタ (図のラベル A および B) を介して DECserver 200 に接続されています。

図 12-2 複数アドレスの LAT 構成: 1 つの LAN にバージョンが混在する LAT ノードがある場合

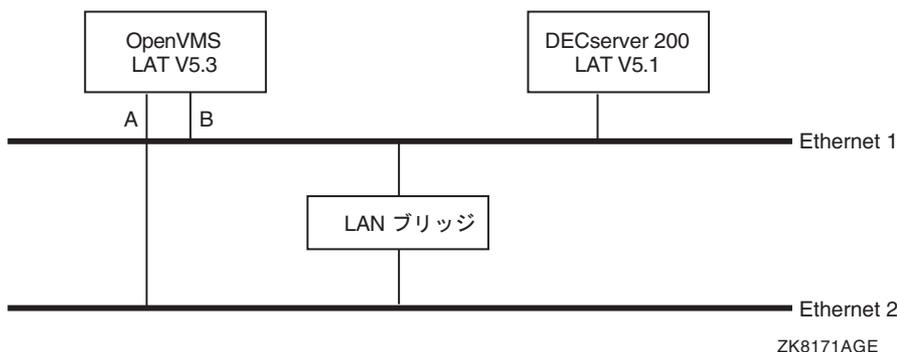


DECserver 200 と OpenVMS システムの間で LAT 接続が開始されると、LAT ソフトウェアは、A と B の両方のアダプタが LAT 仮想サーキットに使用できると判断します。一方のアダプタは 1 次通信パスとして選択され、もう一方は 1 次通信パスに障害が起こった場合に備えておきます。

たとえば、ユーザが DECserver 200 から OpenVMS システムに接続すると、OpenVMS システムは、パスが 2 つあるが、そのうちアダプタ B を 1 次通信パスとして使用することを決定します。ユーザが、OpenVMS システムから大量の出力を生成するプログラムを実行しているとき、出力中にアダプタ B が何らかの理由で障害を起こした場合には、LAT ソフトウェアは、アダプタ A を介して、OpenVMS システムから DECserver への通信を続行しようとします。

図 12-3 「複数アドレスの LAT 構成: 2 つの LAN にバージョンが混在する LAT ノードがある場合」に、ブリッジされた 2 つの LAN を示します。ただし、この構成は、図 12-2 「複数アドレスの LAT 構成: 1 つの LAN にバージョンが混在する LAT ノードがある場合」に示した構成と同じ特徴を示しています。

図 12-3 複数アドレスの LAT 構成: 2 つの LAN にバージョンが混在する LAT ノードがある場合



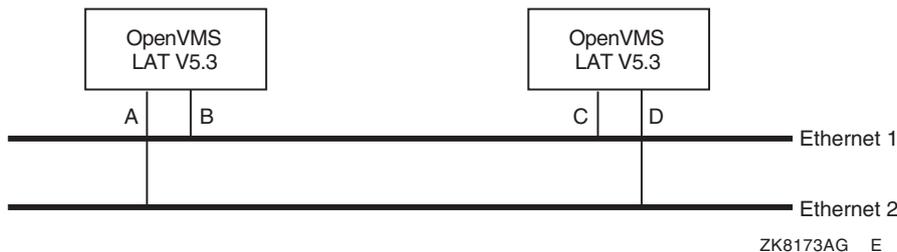


注意:

図 12-3 「複数アドレスの LAT 構成: 2 つの LAN にバージョンが混在する LAT ノードがある場合」のイーサネット 2 は、FDDI ネットワークであると考えられます。LAT ソフトウェアは、各アダプタを同じコストがかかる 2 地点間通信とみなし、FDDI コントローラを全く同様に扱います。ただし、大容量のバッファのサポートについての詳細は、12.3.3 項「イーサネット/FDDI 構成での大容量のバッファ」を参照してください。

図 12-4 「複数アドレスの LAT 構成: 2 つの LAN にバージョン 5.3 の LAT ノードがある場合」の構成の例では、2 つの OpenVMS システム間で作成された仮想サーキットには、2 つのパスがあります。コントローラ B と C、および A と D を介するパスです。1 つのパスで障害が起こった場合、仮想サーキットはもう一方のパスを介して続行されます。両方のパスに障害が起こった場合には、仮想サーキットは結果として時間切れになります。

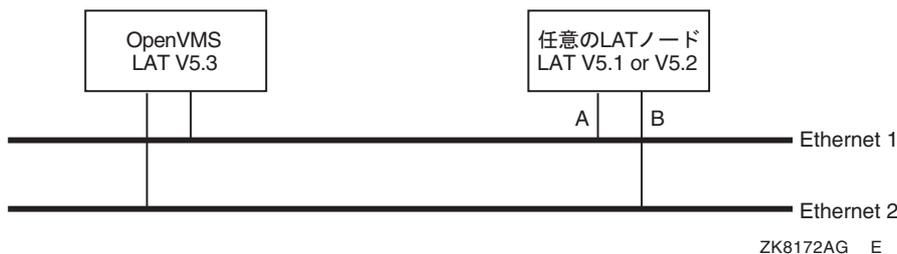
図 12-4 複数アドレスの LAT 構成: 2 つの LAN にバージョン 5.3 の LAT ノードがある場合



12.3.2.2 サポートされない構成

バージョン 5.3 の LAT ソフトウェアを実行する OpenVMS システムを使用するために、ネットワークを構成する場合には、図 12-5 「サポートされない複数アドレスの LAT 構成」に示すような構成は避けてください。

図 12-5 サポートされない複数アドレスの LAT 構成



この図に示したような構成は、予期しない結果となり機能しなくなります。ネットワーク環境では、LAT のバージョン 5.1 および 5.2 のノードは、1 つの論理 LAN アドレスだけしか持つことができません。図 12-5 「サポートされない複数アドレスの LAT 構成」の構成はこの規約に違反します。図 12-4 「複数アドレスの LAT 構成: 2 つの LAN にバージョン 5.3 の LAT ノードがある場合」に示す構成は有効です。

12.3.2.3 論理 LAT リンクの作成

LAT ソフトウェアはすべてのパスを同じコストがかかる、2 地点間通信とみなします。LAT ソフトウェアは最高で 8 個の LAN アダプタを同時にサポートできます (さらに、すべてのコントローラを同じ論理 LAN に接続することもできます)。パス障害に備えて、最大数のアダプタを用意するためには、各論理リンクを作成してから、LAT ノード状態を `SY$MANAGER:LAT$SYSTARTUP.COM` で ON に設定してください。

たとえば、システムに 2 つの FDDI アダプタ (FCA0 と FCB0) を持つイーサネット・アダプタ (デバイス ESA0) が 1 つあり、システム管理者がすべてのアダプタを介して LAT を実行することを選択した場合、`LAT$SYSTARTUP.COM` ファイルには、次のコマンドが含まれます。

```

$!
$! Create each logical LAT link with a unique name and
$! unique LAN address (forced with /NODECNET).
$!
$ LCP CREATE LINK ETHERNET /DEVICE=ESA0 /NODECNET
$ LCP CREATE LINK FDDI_1 /DEVICE=FCA0 /NODECNET
$ LCP CREATE LINK FDDI_2 /DEVICE=FCB0 /NODECNET
$!
$! Turn on the LAT protocol.
$!
$ LCP SET NODE /STATE=ON

```



重要:

リンクが作成される前に LATCP コマンド SET NODE /STATE=ON が入力されると、ランダムな LAT\$LINK または省略時の設定の LAT\$LINK が、一方の LAN アダプタ上に作成されます。どちらの LAN アダプタが選択されるかを予測する方法はありません(システム構成に依存します)。したがって、LAT を起動する前にすべての LAT リンクを作成してください。

各論理リンクは、必ず /NODECNET 修飾子を使用して作成してください。この方法で作成すれば、複数のアダプタが DECnet スタイルのアドレスを使用しようとした際に、リンク作成障害を回避できます。同じアドレスを持つ同じ論理 LAN に複数の LAN アダプタを接続することは LAN 規約に違反し、LAT や他のプロトコルの障害の原因となります。

LAT プロトコルの開始後、論理 LAT データリンクを作成することができます。既存の仮想サーキットは、新しく作成された論理リンクが使用できるようになると、これを使用して、新しいパスを探そうとします。ただし、この時点ではリンクを作成しないでください。この間、LAT プロトコルは既存の仮想サーキットを使用し、この新しく作成されたデータリンクを介して、新しいパスを見つけようとするため、新しいパスが見つかる前に仮想サーキットに障害が起こる可能性があります。

12.3.2.4 パスの発見

OpenVMS LAT ソフトウェアはディレクトリ・サービスと送信請求を組み合わせて使用して、各仮想サーキットのパスを取得します。仮想サーキットの開始時にパスの発見を早めるには、LAT サービスおよびノード・データベースを保守するために複数の LAN アダプタを持つシステムを、次のように構成してください。

- 外部への LAT 接続を有効にする。
- ユーザ・グループとサービス・グループに同じグループ・コード・マスクを使用する。

外部への接続を無効にして、サービスおよびノード・データベースなしで実行されている OpenVMS システムは、各仮想サーキットに複数のパスがあっても実行できます。これらのパスは、LAT 送信請求プロセスで発見されますが、長い時間がかかるでしょう(すべてのパスが発見される前に、仮想サーキットで障害が起こる可能性があります)。

12.3.2.5 LAT パラメータの変更

万一、パスに障害が起こった場合は、OpenVMS LAT ソフトウェアが別の有効なパスを見つけるのに時間がかかります(時間は遠隔ノードがアクセスしているアダプタの数によって異なります)。したがって、有効になる可能性のある LAT マスタ・ノード上の次の LAT パラメータを変更してください。

- 再伝送制限 - 省略時の値は 8。
LAN アダプタの最大数の 8 倍に設定します。たとえば、LAN 上の OpenVMS システムに 3 つのアダプタがある場合には、各 LAT マスタ・ノードの再伝送制限を 24 (3 * 8) に制限しなければなりません。

- キープアライブ・タイマ - 省略時の値は 20 秒。
ほとんどの場合には、省略時の値で十分ですが、30 ~ 40 秒に延ばす必要がある場合もあります。

複数のアダプタを介すことにより、LAT バージョン 5.1 または LAT バージョン 5.2 のマスタ・ノードに仮想サーキットの実行を続けさせることはできませんが、これらのノードへの接続が失敗することもあります。

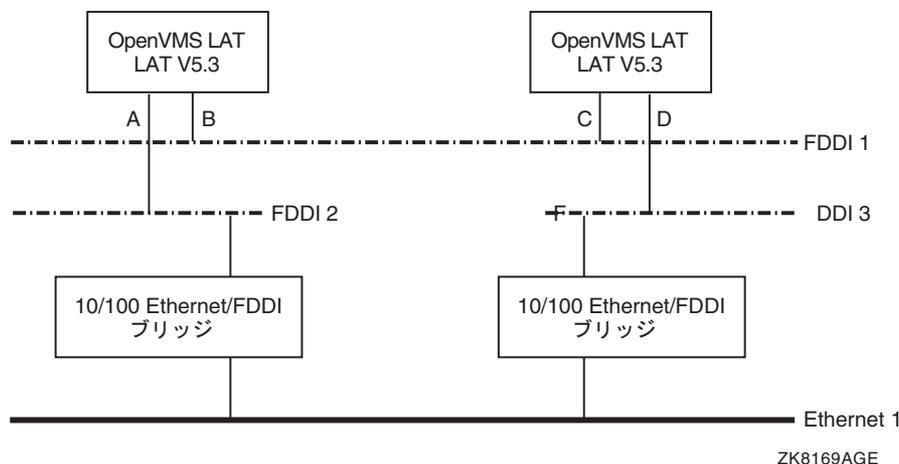
LAT バージョン 5.2 および LAT バージョン 5.1 のマスタ・ノードは、サービスを提供する LAT ノードへの複数のパスを識別することができません。これらのマスタ・ノードは、一度に 1 つの遠隔アドレスを介してだけ、そのようなノードと通信できます。したがって、LAT バージョン 5.1 またはバージョン 5.2 を実行している LAT マスタ・ノードが、サービスを提供している LAT バージョン 5.3 の遠隔ノードに接続しようとした時に LAN パスに障害が起こった場合には、LAT バージョン 5.3 のノードは時間内にこの障害を見つけ出すことができず、LAT マスタ・ノードは時間切れで接続を切断する可能性があります。再伝送制限をできるだけ高く設定することによって、この障害を部分的には解決することができます。

また、1 次パスに障害が起きたとき、サービスを提供している LAT バージョン 5.3 のノードが、仮想サーキットが完全にアイドル状態であると判断した場合には、代替パスのいずれかを使用しようとする試みは行われません (前述の LAT バージョン 5.2 および 5.1 の制約を参照)。したがって、複数の LAN アダプタが古い LAT のインプリメンテーションで動作していても、OpenVMS オペレーティング・システムのバージョン 7.0 以降にアップグレードして、LAT バージョン 5.3 プロトコルを取得する必要があります。これにより、このタイプの問題が解決されます。このタイプの問題は、アイドル状態にある接続だけに影響します。このような状況の例としては、昼休みや終業時刻にすべてのユーザが同時にシステムを離れた場合などがあります。

12.3.3 イーサネット/FDDI 構成での大容量のバッファ

OpenVMS LAT ソフトウェアは、FDDI コントローラを介した仮想サーキットで、**大容量のバッファ**を使用しようとしています。この機能は、代わりの仮想サーキット・パスがイーサネットを介さなければならない場合に、障害を起こすことがあります。図 12-6 「LAT FDDI リングおよび大容量のバッファ」に、障害を起こす構成の例を示します。

図 12-6 LAT FDDI リングおよび大容量のバッファ



この図では、コントローラ B および C により作成されるパスを介することにより、大きなパケットを使用して、2 つの OpenVMS システムが通信することも可能です。**大きなパケット**とは、1500 バイトを超えるデータのことです (イーサネット・メッセージは最大 1500 バイトのデータを含むことができます)。コントローラ B および C により作成されるパスに障害が起こった場合には、A および D によって作成されるパスを介して通信を続行することはできません。

コントローラ A および D により作成されるバスは、イーサネット LAN セグメントを通過します。10/100 ブリッジを介して経路指定されるメッセージは、イーサネット・メッセージの最大サイズより大きくすることはできません。OpenVMS LAT ソフトウェアはこのタイプの構成を常に検出できるとは限らないため、障害が起こる可能性があります。

前述の構成の問題を回避するには、次の 2 つの方法があります。最も簡単な方法は、イーサネット・アダプタを使用して、論理 LAT リンクを作成する方法です (いずれかのシステムにイーサネット LAN アダプタがある場合)。この方法では、メッセージ・サイズ折衝により、イーサネット・メッセージの最大サイズを超えることはありません。

どちらのシステムにもイーサネット・コントローラがない (したがって、最初の方法が使用できない) 場合は、新しい LATCP コマンド修飾子/[NO]LARGE_BUFFER を使用して、大容量のバッファ・サポートの使用を無効にします。たとえば、次のように入力します。

```
$ MCR LATCP SET NODE/NOLARGE_BUFFER
```

SET NODE/NOLARGE_BUFFER コマンドは、すべての論理 LAT リンクを作成した後、LAT ノードを起動する前に使用するようになります。たとえば、次に示す LAT\$SYSTARTUP.COM のコマンドの順序に注意してください。

```
$!  
$! Create each logical LAT link with a unique name and  
$! unique LAN address (forced with /NODECNET).  
$!  
$ LCP CREATE LINK FDDI_1 /DEVICE=FCA0 /NODECNET  
$ LCP CREATE LINK FDDI_2 /DEVICE=FCB0 /NODECNET  
$!  
$! Don't use large buffer support (force packet  
$! sizes to be no larger than what Ethernet can  
$! support).  
$!  
$ LCP SET NODE /NOLARGE_BUFFER  
$!  
$! Turn on the LAT protocol.  
$!  
$ LCP SET NODE /STATE=ON
```

12.4 LATCP ユーティリティについて

LATCP (LAT 制御プログラム) ユーティリティは、OpenVMS システム上の LAT ソフトウェアの構成と制御を行うためのユーティリティ・プログラムです。LATCP のコマンドにより、LAT プロトコルをインプリメントする LAT ドライバの停止と起動、および OpenVMS ノードの LAT 特性の変更や表示を行います。

LATCP ユーティリティにより、システムをサービス・ノードとして設定し、1 つ以上の資源 (サービス) をローカル・エリア・ネットワーク (LAN) 上の他のシステムのユーザから利用可能にすることができます。

さらに、LAN 上の他のシステムのサービスをローカル・システムのユーザが利用できるように設定することも可能です。この場合、システムはターミナル・サーバのように動作して、他のノード上のサービスへの接続を行う複数のユーザ・セッションを同時に管理します。

LATCP を使用すると、システムを外部からのアクセスだけをサポートするように設定することができます。また、外部へのアクセスだけを可能にしたり、両方向のアクセスを可能にすることもできます。さらに、両方向のアクセスともサポートしないようにシステムを設定することもできます。

外部へのアクセスをサポートするように設定されていると、LAT ソフトウェアは LAT サービスとノードのデータベースを管理します。このソフトウェアは、ノード上で外部へのアクセスを可能にするとデータベースを構築します。さらに、LAT の**サービスの通知** (LAT サービス・ノードから送信されるマルチキャスト・メッセージ) の収集を開始し、これらのサービス通知に基づいてデータベースを構築します。このデータベース内に登録されているサービスおよ

びノードを表示したり、データベースのサイズを設定したりするときにも LATCP ユーティリティを使用します。スタンドアロン・システムのようにオーバーヘッドの増加をある程度許容できるシステムでは、外部へのアクセスを可能にしてください。

LATCP ユーティリティは、次の作業に使用します。

- ノードの動作特性とサービスを指定する。
- LAT ポート・ドライバ (LTDRIVER) をオンまたはオフにする。
- ネットワークの LAT サービスとサービス・ノードの状態を表示する。
- LAT ノード上で作成されたリンクの状態を表示する。
- LAT ノードの状態を表示する。
- LAT カウンタの値の表示とリセットを行う。
- LAT ポートの作成、削除、および管理を行う。
- LATCP コマンドの再呼び出しを行う (この機能により、一度入力したコマンドを再入力しないで再び実行することができる)。
- サブプロセスを作成する (LATCP を終了しなくても DCL のコマンドを実行することができる)。

LAT プロトコルを使用すると、LAN 上の LAT ターミナル・サーバまたはサービス・ノードに接続されたプリンタおよび他の非同期デバイスに各ユーザがアクセスすることができるように、ローカル・ノード上で LAT アプリケーション・ポートを設定することができます。その場合、リモート・デバイスは適切に構成する必要があります。

12.4.1 LATCP の起動と終了

LATCP を起動するためには、次のコマンドを入力します。

```
$ RUN SYS$SYSTEM:LATCP
LATCP>
```

LATCP> プロンプトが表示されている状態で、LATCP の各コマンドを入力することができます。LATCP を終了するためには、このプロンプトに対して EXIT と入力するか、Ctrl/Z を押します。また、次の例のように DCL 文字列代入文を使用して、単一の LATCP コマンドを実行することもできます。

```
$ LCP ::= $LATCP
$ LCP SET NODE/STATE=ON
```

LATCP は SET NODE コマンドを実行したのち、DCL に制御を返します。

12.4.2 LATCP コマンド

表 12-1 「LATCP コマンド」は、LATCP の各コマンドについてまとめています。

表 12-1 LATCP コマンド

コマンド	機能
ATTACH	現在のプロセスから指定されたプロセスに制御を移す。
CREATE LINK	LAT データ・リンクを作成する。
CREATE PORT	ローカル・ノード上に論理ポートを作成する。
CREATE SERVICE	サービス・ノード上にサービスを定義する。
DEFINE/KEY	コマンド文字列をキーボード上のファンクション・キーに割り当てる。
DELETE LINK	ノードから LAT データ・リンクを削除する。
DELETE PORT	アプリケーション・ポートまたは専用ポートを削除する。
DELETE QUEUE_ENTRY	ローカル・ノードからキュー登録された接続要求を削除する。

表 12-1 LATCP コマンド (続き)

コマンド	機能
DELETE SERVICE	サービス・ノード上のサービスを削除する。
EXIT	DCL コマンド・レベルに戻る。
HELP	LATCP コマンドのヘルプ・テキストを表示する。
RECALL	以前に入力された LATCP コマンドを再び呼び出して、そのコマンドを再度実行できるようにする。
REFRESH	スクリーンの内容をリフレッシュする。たとえば、スクリーンの表示が他のソースからの出力によって上書きされた場合に使用する。
SCROLL	スクロールして画面から消えた情報の検索を可能にする。
SET LINK	LAT データ・リンクの特性を変更する。
SET NODE	ノードの LAT 特性を指定する。
SET PORT	ノード上の論理ポートをターミナル・サーバ上のリモート・デバイスまたは遠隔 LAT サービス・ノード上の特殊アプリケーション・サービスのいずれかにマップする。
SET SERVICE	サービス特性を変更する。
SHOW LINK	ノード上に存在するリンクの特性を表示する。
SHOW NODE	ノードの特性を表示する。
SHOW PORT	ポートの特性を表示する。
SHOW QUEUE_ENTRY	ローカル・ノードでキュー登録された要求または入力に関する情報を表示する。
SHOW SERVICE	ローカル・ノードが認識している LAT サービスの特性を表示する。
SPAWN	サブプロセスを作成する。
ZERO COUNTERS	ローカル・ノードで保持されているノード・カウンタ、サービス・カウンタ、リンク・カウンタをリセットする。

LATCP の各コマンドと指定可能な修飾子についての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

12.5 LAT プロトコルのスタートアップ

システム管理者として、LAT プロトコルをスタートアップし、ローカル・ノードをサービス・ノードとして構成します。そのためには、コマンド・プロシージャ `SYS$STARTUP:LAT$STARTUP` を実行します。このプロシージャは、次の 2 つのプロシージャを実行します。

1. LAT\$CONFIG.COM

LAT ターミナル・ドライバである `LTDRIVER` をロードして、LATACP プロセスを作成する。

2. LAT\$SYSTARTUP.COM

LAT 特性を定義する LATCP のコマンドを実行する。

作業方法

システムのブート時に必ず LAT プロトコルが起動されるようにするためには、以下に示すように、汎用のサイト別スタートアップ・コマンド・プロシージャにこのプロシージャを実行するコマンドを追加します。このコマンド・プロシージャについての詳細は『OpenVMS システム管理者マニュアル(上巻)』を参照してください。そこでは、オペレーティング・システムでこのコマンド・プロシージャを特定するためのファイル指定などを示しています。

システムがブートするたびにローカル・ノードを LAT サービス・ノードとして設定し、システム上で LAT プロトコル・ソフトウェアを起動するには、汎用のサイト別スタートアップ・コマンド・プロシージャに次の行を追加します。

```
$ @SYS$STARTUP:LAT$STARTUP.COM
```

汎用のサイト別スタートアップ・コマンド・プロシージャがこのコマンドを実行すると、LAT\$STARTUP.COM が呼び出されます。LAT\$STARTUP.COM は、コマンド・プロシージャの LAT\$CONFIG および LAT\$SYSTARTUP を呼び出します。

LAT\$STARTUP を呼び出すコマンド行に次の引数を指定して、ローカル・ノード独自の LAT 特性を指定することができます。プロシージャはこれらの引数を LAT\$SYSTARTUP.COM に渡し、指定された LAT 特性を定義します。

```
$ @SYS$STARTUP:LAT$STARTUP "P1" "P2" "P3" "P4" "P5"
```

P1 から **P5** にパラメータを渡すよりも、できるだけ LAT\$SYSTARTUP.COM を直接変更するようにしてください。P1 から P5 を使用する場合は、次の表を参照して各パラメータの意味を理解してください。

引数	形式	意味
P1	サービス名	サービスの名前。サービス・ノードがクラスタ化されている場合には、サービス名としてクラスタ別名を使用する。サービス・ノードが独立している場合には、DECnet ノード名を使用する。LAT\$SYSTARTUP.COM は (LATCP CREATE SERVICE コマンドを使用して)、引数 P1 に指定されたサービス名をノードに割り当てる。
P2~P4	以下のいずれか /IDENTIFICATION= " 文字列 "	LAT\$SYSTARTUP.COM は (LATCP SET NODE コマンドを使用して)、この引数に指定された値に基づいて LAT ノード特性を割り当てる。 ノードおよびそのノードからローカル・エリア・ネットワーク (LAN) 上に通知されるサービスの説明。この値を省略すると、論理名 SYS\$ANNOUNCE で定義された文字列が割り当てられる。指定する文字列は 5 組の二重引用符で囲む必要がある。 例: "/IDENTIFICATION=" - """"Official system center""""
	/GROUPS=(ENABLE= グループのリスト)	サービス・ノードとの接続を許可するターミナル・サーバ・グループを設定する。省略時の設定では、グループ 0 が接続可能になる。
	/GROUPS=(DISABLE= グループのリスト)	指定されたターミナル・サーバ・グループを接続可能に設定されているグループから削除する。後に示す例にあるように、修飾子 /GROUPS の後ろに、接続を許可するグループと接続を禁止するグループの両方を同時に指定することができる。
P5	CREATE SERVICE コマンドで使用する修飾子のいずれか。	LAT\$SYSTARTUP.COM は LATCP CREATE SERVICE コマンドを使用し、この引数で指定された値に基づいてサービス特性を割り当てる。修飾子 /IDENTIFICATION、/LOG、および /STATIC_RATING を指定することができる。次の例に示すように複数の修飾子を指定する。 "/IDENTIFICATION=" - """"Official system node"""" - /STATIC_RATING=250"

LAT ネットワークに関する以下のいずれかの作業を行う場合は、LAT\$SYSTARTUP.COM を変更する必要があります。変更内容については、12.6 項「LAT 特性のカスタマイズ」を参照してください。

- LAT プリンタを設定する。
- 特殊アプリケーション・サービスを定義する。
- 外部への接続が可能になり、SET HOST/LAT コマンドをサポートするようにノードを設定する。

LATCP コマンドとその修飾子に関する全情報については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

例

次のコマンドは、クラスタ OFFICE に属するサービス・ノード NOE 上にサービス OFFICE を定義します (図 12-1 「LAT ネットワークの構成例」を参照)。

```
§ @SYS$STARTUP:LAT$STARTUP OFFICE
```

12.6 LAT 特性のカスタマイズ

ローカル・ノードの特殊な LAT 特性を定義する場合は、サイト別コマンド・プロシージャ SYS\$MANAGER:LAT\$SYSTARTUP.COM を編集します。このコマンド・プロシージャには、LAT 特性を定義するための LATCP のコマンドが含まれています。LAT\$SYSTARTUP.COM は、コマンド・プロシージャ LAT\$STARTUP を実行すると呼び出されます。12.5 項「LAT プロトコルのスタートアップ」で説明したように、通常は LAT\$STARTUP.COM を汎用のサイト別スタートアップ・コマンド・プロシージャから実行します。

ローカル・ノードを会話型ターミナルからの接続だけをサポートする LAT サービス・ノードとして設定する場合には、LAT\$SYSTARTUP.COM を変更する必要はありません。12.5 項「LAT プロトコルのスタートアップ」で説明したように、コマンド・プロシージャ SYS\$STARTUP:LAT\$STARTUP を呼び出すときにパラメータを指定することにより、サービス名および他の特性を割り当てることができます。

ただし、LAT\$SYSTARTUP.COM を編集すると、ローカル・ノードの LAT 特性をカスタマイズするための LATCP のコマンドを追加することができます。カスタマイズ可能な項目は次のとおりです。

作業	参照箇所
複数のサービスの定義	12.6.1 項「付加サービスの定義」
特殊アプリケーション・サービスおよびプリンタのための論理ポートの設定	12.6.2 項「ポートの設定」
外部からの接続要求のキュー登録	12.6.3 項「外部からの接続要求のキュー登録」
外部への LAT 接続を可能にするることによる SET HOST/LAT コマンドのサポート	12.6.4 項「外部への LAT 接続の許可」
ノード特性の変更 ¹	12.6.5 項「LAT\$SYSTARTUP.COM プロシージャの変更例」

1 例として、SET NODE コマンドおよび SET LINK コマンドによる特殊なサービス通知または LAN リンクの割り当てがある。



重要:

コマンド・プロシージャの LAT\$STARTUP.COM および LAT\$CONFIG.COM は変更しないでください。これらのプロシージャは、LAT プロトコルを正しく稼働させるために必要な機能を実行するために弊社が提供したものです。コマンド・プロシージャを変更するのは、LAT\$SYSTARTUP.COM でサイト固有の LAT 特性を定義する場合に限ります。

LAT\$SYSTARTUP.COM には LATCP ユーティリティのコマンドを追加するだけにしてください。また、コマンドの順序は、テンプレート・ファイル SYS\$MANAGER:LAT\$SYSTARTUP.TEMPLATE 内の順序に従ってください。12.6.5 項「LAT\$SYSTARTUP.COM プロシージャの変更例」に、LAT\$SYSTARTUP プロシージャの変更例を示します。『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』には、LAT\$SYSTARTUP.COM に含むことができるすべての LATCP コマンドの全情報が示されています。

12.6.1 付加サービスの定義

弊社が提供する LAT\$SYSTARTUP.COM プロシージャは、1 つのサービスを定義します。このプロシージャで定義するサービスは、ユーザが汎用のシステム環境にアクセスするために使用する 1 次サービス、あるいはデータ・エントリ・プログラムやオンライン・ニュース・サービスのような特殊なアプリケーション・サービスのどちらにすることもできます。

Alpha システムの場合、12.6.2.3 項「制限されたサービスの設定」で説明するように、決まった数の LTA デバイスに制限されたサービスを定義することもできます。

LAT\$SYSTARTUP.COM プロシージャが定義するサービスの名前はローカル・ノードの名前と同じになります。ただし、12.5 項「LAT プロトコルのスタートアップ」で説明したように、@SYS\$STARTUP:LAT\$STARTUP.COM コマンドの引数として一意のサービス名を指定すれば、その名前が割り当てられます。

作業方法

LAT\$SYSTARTUP.COM によって定義するサービスのほかにもサービスを定義する場合は、CREATE SERVICE コマンドを使用します。このコマンドは、LAT\$SYSTARTUP.COM に追加することができます。アプリケーション・サービスを定義する場合、そのサービスにはできるだけアプリケーション・プログラムの名前を割り当てるようにしてください。LATCP ユーティリティの CREATE SERVICE コマンドについての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

例

次の例は、ローカル・ノード上にアプリケーション・サービス NEWS を定義します。

```
$ LCP := $LATCP
$ LCP CREATE SERVICE /APPLICATION NEWS
```

12.6.2 ポートの設定

弊社が提供する LAT\$SYSTARTUP.COM ファイルには、サービス・ノード上に論理ポートを作成したり、それらのポートをターミナル・サーバ・ノード上の物理ポートやサービスに対応づけたりするためのサンプル・コマンドが含まれています。これらのポートは、アプリケーション・サービスおよび遠隔プリンタに使用できます。

作業方法

ポートを作成する場合は、LAT\$SYSTARTUP.COM ファイルのサンプル・コマンドの前にある感嘆符 (!) を削除します。あるいは、必要に応じてサンプル・コマンドと同様の CREATE PORT コマンドおよび SET PORT コマンドをファイルに追加します。LATCP ユーティリティのコマンド CREATE PORT および SET PORT については、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』を参照してください。



注意:

アプリケーション・ポートおよび専用ポートは、LATCP ユーティリティのコマンド SET NODE/STATE=ON を実行した**後**で作成するようにしてください。それにより、非ページング・プール・メモリの使用量が最小限に抑えられ、ポートを重複して作成することがなくなります。

LCP CREATE PORT LTA5001:/APPLICATION などのコマンドを使用してアプリケーション・ポートを作成しようとしているとき、次のようなエラーが発生する場合があります

```
%LAT-W-CMDERROR, error reported by command executor  
-SYSTEM-F-DUPLNAM, duplicate name
```

このエラーは、作成しようとしている LAT アプリケーション・ポートが他のアプリケーションによってすでに作成されていることを示します。このポートは、LATCP 自身が作成したものである場合もあります。LATCP のポート LATCP\$MGMT_PORT は、LTDRIIVER と通信を行うときに使用されます。

このエラーが発生しないようにするためには、アプリケーション・ポートまたは専用ポートを作成するどのコマンドを実行するよりも前に、SET NODE/STATE=ON コマンドを実行します。LATCP の SET NODE/DEVICE_SEED コマンドを使用する方法もあります。SET NODE/DEVICE_SEED コマンドについての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』を参照してください。

12.6.2.1 プリンタの設定

プリンタ用のポートを設定する場合は、以下の作業も同時に行う必要があります。

1. プリンタ用のスプール出力キューを作成する。
2. 作成したスプール出力キューを起動するコマンドを、他のキューを起動するスタートアップ・コマンド・プロシージャまたは汎用のサイト別コマンド・プロシージャに追加する。

これらの作業については、『OpenVMS システム管理者マニュアル (上巻)』で説明しています。

12.6.2.2 特殊アプリケーション・サービスの設定

特殊アプリケーション・サービスを確立するためには、LAT ポートを定義するときに /DEDICATED 修飾子を指定します。サービスの接続先のアプリケーション・プログラムで、同じ専用ポートを定義する必要があります。たとえば、次のコマンドは NEWS というアプリケーション・サービスのためのポートを設定します。

```
$ LCP ::= $LATCP  
$ LCP CREATE PORT LTA333:/DEDICATED  
$ LCP SET PORT LTA333:/SERVICE=NEWS
```

アプリケーション・サービスを LAT ネットワーク上のユーザ・ターミナルから利用可能にするためには、アプリケーション・プログラムを起動する必要があります。通常は、アプリケーション起動のためのコマンドを SYLOGIN.COM に追加します。

12.6.2.3 制限されたサービスの設定

専用ポートを持ったアプリケーション・サービスを使用すると、システムが提供するプロセスの制御下にある LTA デバイス (LAT ターミナルなど) を、あらかじめ決まった数だけ作成できます。ただし、このような環境では、専用 LTA デバイスがシステム・ログイン・イメージ (LOGINOUT.EXE) を実行する方法がないので、ユーザはサービスにログインできません。

制限されたサービスを作成して、このサービスに関連する、あらかじめ決まった数の LTA デバイスにユーザがログインできるようにすることが可能です。LTA デバイスがすべて使用中の場合、LAT ソフトウェアはこのサービスへの追加接続要求を拒否して、“service in use” というエラー・メッセージを表示します。このようにして制限されたサービスを作成すると、システム

上の LAT ユーザの数を制限できます。ただし、ユーザが制限されたサービスに接続する際の LTA デバイスの割り当て方は制御できません。

次の例は、2 つの LTA デバイスを持つ制限されたサービスを設定します。

```
$ LCP ::= $LATCP
$ LCP CREATE SERVICE /LIMITED RESTRICTED
$ LCP CREATE PORT LTA100 /LIMITED
$ LCP CREATE PORT LTA101 /LIMITED
$ LCP SET PORT LTA100 /SERVICE=RESTRICTED
$ LCP SET PORT LTA101 /SERVICE=RESTRICTED
```

ユーザが RESTRICTED という名前の制限されたサービスに接続を試みると、LAT ソフトウェアは LTA100 または LTA101 の内、最初に使用できる方を選択して、接続を完了します。これでユーザはシステムにログインできます。別のユーザがサービスに接続しようとする、もう一方の LTA デバイスに割り当てられます。このユーザは 2 番目のシステムにログインできます。RESTRICTED という名前の制限されたサービスに関連するこの 2 つのデバイスが両方とも使用中であれば、この制限されたサービスへのこれ以降の接続要求は拒否されて、エラー・メッセージ “service in use” が表示されます。

ユーザがシステム (LTA100 または LTA101) からログアウトしても、LTA デバイスは削除されません。制限されたサービスへの次の接続要求を受け付けられるように再設定されます。

12.6.3 外部からの接続要求のキュー登録

省略時の設定では、制限されたサービスまたはアプリケーション・サービスへの外部からの接続要求はキューに登録されます。つまり、ユーザが (順方向キュー登録が可能なターミナル・サーバ・ポートを使用するか、または DCL コマンド SET HOST/LAT/QUEUE を入力して) 制限されたサービスまたはアプリケーション・サービスへの接続を要求すると、サービスでポートが使用できない場合には、LAT ソフトウェアはこの接続要求を拒否するのではなくキューに登録します。

作業方法

外部からの接続要求をキューに登録するサービスは、次のような設定および管理を行うことができます。

- LATCP コマンド SHOW SERVICE を使用して、サービスでキュー登録が可能かどうかを確認する。
- キュー登録が禁止されていれば、SET SERVICE /QUEUE コマンドを使用して可能にする。
- SET NODE /QUEUE_LIMIT=*n* コマンドをローカル・ノードで使用して、空きキュー・スロットの数を制御する (*n* は 0 から 200)。
- SHOW QUEUE_ENTRY [エン트리 ID] コマンドを使用して、ローカル・キューのエントリを表示する。
- DELETE QUEUE_ENTRY [エン트리 ID] コマンドを使用して、ローカル・キューからエントリを削除する。

キュー登録要求のサポートに使用する LATCP コマンドおよび修飾子の詳細は『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』を参照してください。

例

次は、システム上でのキュー登録の許可方法を示す例です。

```
$ LCP ::= $LATCP
$ LCP SET SERVICE /QUEUE
```



注意:

システムがキュー登録された接続要求を処理するように構成されている場合は、次の設定を行ってキュー接続の失敗を防ぐ**必要があります**。

- 外部からおよび外部への接続を許可する。
- ユーザ・グループ・コードおよびサービス・グループ・コードを同一にする。

12.6.4 外部への LAT 接続の許可

各ノードの省略時の設定では、外部への LAT 接続は不可能になっています。各ユーザが SET HOST/LAT コマンドによりローカル・ノードから LAT 接続を確立することを許可する場合は、外部への接続が可能になるように LAT\$SYSTARTUP.COM を変更します。SET HOST/LAT コマンドを使って外部への LAT 接続を行う方法についての詳細は、『OpenVMS DCL ディクショナリ』にあるこのコマンドの説明を参照してください。

外部への接続を可能にするコマンドは、弊社が提供する LAT\$SYSTARTUP.COM ファイルに含まれています。実行したいコマンドがあれば、その前に付いている感嘆符 (!) を削除します。あるいは、必要に応じて同様のコマンドをこのファイルに追加します。詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』の中の LATCP ユーティリティのパートにある、SET NODE コマンドの修飾子 /CONNECTIONS および /USER_GROUPS の説明を参照してください。

SET HOST/LAT の性能を最適化し、ポートの性能を向上させるためには、システム・パラメータ TTY_ALTYPAD の値を 1,500 に設定してリポートします。

ローカル・ノードを外部からの接続だけが可能なサービス・ノードとして設定する場合は、LAT\$SYSTARTUP.COM を変更する必要はありません。ただし、次の場合は LAT\$SYSTARTUP.COM を変更します。

- ノード上で複数のサービスを定義する場合
- 特殊アプリケーション・サービスを定義する場合
- LAT プリンタを設定する場合
- 外部への LAT 接続を可能にして、ローカルのノードをターミナル・サーバ・ノードとして機能させる場合
- ノード特性を変更する場合 (たとえば、特殊なサービス通知や LAN への接続を割り当てる場合)

12.6.5 LAT\$SYSTARTUP.COM プロシージャの変更例

次に示すのは、LAT\$SYSTARTUP.COM プロシージャを変更した例です。この変更により、サービスの定義、ポートの作成と設定、外部からおよび外部への両方向の接続の許可が行われます。

```

$!
$!   LAT$SYSTARTUP.COM -- LAT Startup Commands Specific to Site
$!
$!   Use this command procedure to customize the LAT characteristics for
$!   the local node. These commands, which should serve as examples,
$!   will set up a LAT service name SYS$NODE and default identification
$!   SYS$ANNOUNCE. The LAT service name and identification will default
$!   to SYS$NODE and SYS$ANNOUNCE unless you specify a service name and
$!   identification as arguments to the command line that invokes
$!   LAT$STARTUP.COM:
$!           $ @SYS$STARTUP:LAT$STARTUP
$!
$!   You can specify other node and service characteristics (such as group
$!   codes) as arguments to this command line, as shown below.
$!
$!           Argument      Function
$!           -----      -
$!

```

```

$!           P1           Name of the service to be created.  If not supplied, a
$!                               service will be created with the same name as the node.
$!
$!           P2,P3,P4     Parameters and qualifiers to the SET NODE command.
$!
$!           P5           Parameters and qualifiers to the SET SERVICE command.
$!                               P5 is only used if P1 is specified.  More than one
$!                               argument may be supplied by enclosing the string in
$!                               quotes.
$!
$! Example: $ @SYS$STARTUP:LAT$STARTUP HAWK "/IDENTIFICATION=" -
$!           """"Development node""""
$!
$! Please review and edit this file for possible additions and deletions
$! that you wish to make.  Future software updates will not overwrite the
$! changes made to this file.
$!
$ required_privileges = "OPER"
$ prev_privs = f$setprv(required_privileges)
$ if .not. f$privilege(required_privileges) then goto no_privileges
$ lcp := $latcp
$!
$! ----- Modify Node Characteristics -----
$!
$ lcp set node 'p2' 'p3' 'p4'
$!
$! Some examples:
$!
$! ** Allow incoming connections only
$!
$! lcp set node /connections=incoming /groups=(enable=(12,40,43,73),disable=0)
$! lcp set node /connections=incoming /groups=enable=(0-255)
$!
$ LCP SET NODE /CONNECTIONS=INCOMING /GROUPS=(ENABLE=(12,40,43,73),DISABLE=0)
$!
$! ** Allow outgoing connections only
$!
$! lcp set node /connections=outgoing /user_groups=enable=(24,121-127)
$! lcp set node /connections=outgoing /user_groups=(enable=0-255) /node_limit=50
$!
$! ** Enable incoming and outgoing connections
$!
$! lcp set node /connections=both /group=enable=(43,73) /user=enable=(44,56)
$! lcp set node /connections=both /group=enable=(0-255) /user=enable=(0-255)
$!
$! ----- Modify Service Characteristics -----
$!
$ if p1 .eqs. ""
$ then
$   lcp create service
$ else
$   lcp create service 'p1' 'p5'
$ endif
$! ----- Start LAT Protocol -----
$!
$ lcp set node /state=on
$!
$! ----- Create and Map Ports -----
$!
$! Some examples:
$!
$! lcp create port lta101: /dedicated
$! lcp create port lta102: /application
$! lcp create port lta103: /application
$! lcp create port /nolog/logical=(name=ln03$mgmt, table=system, mode=executive)
$
$ LCP CREATE PORT LTA1: /NOLOG
$ LCP CREATE PORT LTA20: /NOLOG
$
$! lcp set port lta101: /dedicated /service=graphics

```

```

$! lcp set port lta102: /node=server_1 /port=port_1
$! lcp set port lta103: /node=server_2 /service=laser
$! lcp set port ln03$mgmt: /node=server_3 /service=ln03_printers
$!
$ LCP SET PORT LTA1: /APPLICATION/NODE=TERM_SERVER_1 /PORT=PORT_6
$ LCP SET PORT LTA20: /APPLICATION/NODE=TERM_SERVER_2 /PORT=PORT_6
$!
$exit:
$ prev_privs = f$setprv(prev_privs)
$ exit
$!
$no_privileges:
$ write sys$output "Insufficient privileges to execute LATCP commands."
$ write sys$output "Requires ",required_privileges," privileges."
$ goto exit

```

12.7 LATACP データベースのサイズの管理

OpenVMS ノード上では、LAT ソフトウェアの別のコンポーネントである LAT 補助制御プロセス (LATACP) によって、利用可能なノードとサービスのデータベースが保守されます。このデータベースには、遠隔 LAT ノードからマルチキャストされたノードとサービス、あるいは、ローカル・システムで定義したローカル・ノードと 1 つ以上のローカル・サービスが登録されます。このデータベースのサイズの上限は、システム・パラメータ CTLPAGES の値に依存します。

LATCP コマンドの入力後、次のような応答メッセージを受け取ることがあります。

```

%LAT-W-CMDERROR, error reported by command executor
-LAT-F-ACPNOCPL, insufficient resources - ACP CTL/P1 space limit reached

```

このメッセージは、データベースのサイズが CTLPAGES で指定されている上限に達したことを示します。次にいずれかの方法でこの状態を修正することができます。

- ノードの制限値を低くしてデータベースのサイズを縮小する。ノードの制限値は LATCP の SHOW NODE コマンドによって表示できる。この値を変更する場合は、SET NODE/NODE_LIMIT コマンドを使用する。詳細は『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』を参照。
- ノード上で使用可能になっているユーザ・グループ・コードを減らして、データベースのサイズを縮小する。使用可能になっているユーザ・グループ・コードは、LATCP の SHOW NODE コマンドによって表示できる。その一部を使用不能にするためには、SET NODE/USER_GROUPS=DISABLE コマンドを使用する。詳細は『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』を参照。

この方法を選択した場合は、スタートアップ・コマンド・プロシージャを編集して、システムがリブートするたびに使用可能になるユーザ・グループを変更する必要がある。詳細は 12.6 項「LAT 特性のカスタマイズ」を参照。

- システム・パラメータ CTLPAGES の値を大きくしてデータベースのサイズを拡大する。一般に、CTLPAGES の値を 1 単位大きくすると、データベースにはノードまたはサービスのおよそ 6 件分のデータがさらに登録できるようになる。

CTLPAGES の変更後にその値を有効にするために、システムをリブートする。このとき、CTLPAGES の変更後の値が AUTOGEN のパラメータ・ファイル MODPARAMS.DAT に追加されていることを確認する。システム・パラメータ値の変更については、1.2 項「パラメータ値の標準的な変更方法」を参照。

第13章 特殊処理環境の管理

この章は、主に OpenVMS VAX ユーザ向けです。OpenVMS Alpha と I64 の拡張機能は、他の OpenVMS ドキュメントで説明します。

OpenVMS オペレーティング・システムは以下の特殊環境をサポートします。

- 対称型マルチプロセッシング
- ベクタ処理 (特定の CPU モデルだけで利用可能)

この章では、これらの特殊処理環境の設定と管理について説明します。

この章の内容

この章では、次の作業について説明します。

作業	参照箇所
マルチプロセッシング環境の作成	13.2.1 項 「マルチプロセッシング環境の作成」
マルチプロセッシング環境の監視	13.2.2 項 「マルチプロセッシング環境の監視」
ベクタ処理サポート・コードのロード ¹	13.4.1 項 「ベクタ処理サポート・コードのロード (VAX のみ)」
ベクタ処理システムの構成 ¹	13.4.2 項 「ベクタ処理システムの構成 (VAX のみ)」
ベクタ・プロセスの管理 ¹	13.4.3 項 「ベクタ・プロセスの管理 (VAX のみ)」
ACL によるベクタ・プロセッサへのアクセス制限 ¹	13.4.4 項 「ACL によるベクタ・プロセッサへのアクセスの制限 (VAX のみ)」
ベクタ処理システムに関する情報の入手 ¹	13.4.5 項 「ベクタ処理システムに関する情報の入手 (VAX のみ)」
VAX ベクタ命令エミュレーション機能 (VVIEF) のロード ¹	13.4.6 項 「VAX ベクタ命令エミュレーション機能 (VVIEF) のロード (VAX のみ)」

¹ VAX のみ

さらに、次の項目について説明します。

項目	参照箇所
対称型マルチプロセッシング	13.3 項 「ベクタ処理について」
1 次プロセッサと 2 次プロセッサ	13.1.1 項 「1 次プロセッサと 2 次プロセッサ」
アベイラブル・セットとアクティブ・セット	13.1.2 項 「アベイラブル・セットとアクティブ・セット」
ベクタ処理	13.3 項 「ベクタ処理について」
VAX におけるベクタ処理のサポート ¹	13.3.1 項 「VAX におけるベクタ処理のサポート (VAX のみ)」
VAX ベクタ命令エミュレーション機能 (VVIEF) ¹	13.3.2 項 「VAX ベクタ命令エミュレーション機能 (VVIEF) (VAX のみ)」

¹ VAX のみ

13.1 マルチプロセッシングについて

マルチプロセッシング・システムは、メモリの共通プールにアドレスして同時に命令を実行することが可能な 2 つ以上の CPU から構成されます。

OpenVMS オペレーティング・システムは密結合対称型マルチプロセッシング (SMP) システムをサポートします。密結合 SMP システムでは、すべてのプロセッサがオペレーティング・システムの単一のコピーを実行し、すべてのオペレーティング・システム・コードとシステム資源に対する等しいアクセス権を持っています。OpenVMS SMP は、プロセス優先権に基づいてプロセスが実行される CPU を動的に選択します。

マルチプロセッシング・システムは、独立したシステム、ネットワーク上のノード、または OpenVMS Cluster を構成するメンバとして機能することができます。マルチプロセッシング・システムおよびユニプロセッシング・システムは、同じオペレーティング・システムを実行しません。ただし、マルチプロセッシングは特定の VAX, Alpha, および I64 プロセッサ上でしか行うことができません。1 つのマルチプロセッシング環境を構成するすべてのプロセッサのハードウェアとファームウェアは同じレベルであることが必要です。これは、特定のプロセッサから、システム上の別のプロセッサで実行されていたプロセスの実行スレッドを再開できるようにするためです。

13.1.1 1 次プロセッサと 2 次プロセッサ

マルチプロセッシング・システムでは、コンソール・デバイスに論理的または物理的に付加された 1 つのプロセッサがシステム内の他のプロセッサの起動を担当します。そのようなプロセッサを **1 次プロセッサ** と呼びます。1 次プロセッサ自身は、マルチプロセッシング・システムをブートするコンソール・コマンドのターゲットとなります。1 次プロセッサは、システム全体のオペレーティング・システム環境の定義やメモリの準備のための初期化を単独で行います。さらに、1 次プロセッサはシステム内の計時を担当し、システム時間の管理、およびタイム・キューに登録された各要素がタイムアウトになっていないかどうかの監視を行います。一方、マルチプロセッシング・システム内のこれらの役割を持たないプロセッサを **2 次プロセッサ** と呼びます。

13.1.2 アベイラブル・セットとアクティブ・セット

システムのハードウェア電源投入時の診断テストにパスしたプロセッサの集合を **アベイラブル・セット** と呼んでいます。このとき、各プロセッサがシステムに対してアクティブに関与しているかどうかは問題にされません。1 次プロセッサと 2 次プロセッサは、マルチプロセッシング・システムのアクティブ・セットから構成されます。

アクティブ・セット とは、VAX システム、Alpha システム、または I64 システムを構成するプロセッサのうち、電源投入時の診断テストにパスし、システム動作にアクティブに関与しているプロセッサの集合をいいます。オペレーティング・システムはこれらのセットに含まれる各プロセッサを CPU ID で識別します。CPU ID の値は、DCL やユーティリティの特定のコマンドの構文や表示の中に頻繁に現れます。

13.1.3 プロセッサの機能

マルチプロセッシング・システムはシステム上で実行中のプロセスに特定の機能を提供します。次の機能がサポートされています。

- 1 次
- クォーラム
- 実行
- ベクタ (VAX のみ)

さらに、他の機能を追加したり、削除したりするメカニズムもあります。

実行機能は、CPU の開始動作と停止動作に影響します。

13.2 対称型マルチプロセッシング (SMP) 環境の管理

対称型マルチプロセッシング・システム (SMP) の管理には、マルチプロセッシング環境の作成と監視があります。

13.2.1 マルチプロセッシング環境の作成

マルチプロセッシング・システムの構成要素と特性は、適切なシステム・パラメータを設定することによって、システムのブート時に制御できます。マルチプロセッシング・システムを管理するシステム・パラメータには以下のものがあります。

パラメータ	機能
MULTIPROCESSING	ブート時にオペレーティング・システムにロードする同期イメージを決定する。
SMP_CPUS	ブート時に、アベイラブル・セットからマルチプロセッシング環境に追加するプロセッサを決定する。

以上のようなシステム・パラメータについては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。

ブート時に、アベイラブル・セットの中のプロセッサをアクティブ・セットに追加することができます。また、DCL の START/CPU コマンドを使用すれば、ブート後にプロセッサの追加を行うことができます。アクティブ・セットからプロセッサを削除するときには、DCL の STOP/CPU コマンドを使用します。

SMP Extension License

Alpha システムおよび I64 システムでは、SMP システムを使用している場合には、SMP Extension License を登録する必要があります。このライセンスは Operating System Base License および全 Interactive User ライセンスをアップグレードして、システムのマルチプロセッシング・レベルに合わせます。

SMP Extension License は、既存の Base ライセンスおよび User ライセンスがユニプロセッシング・レベルで提供するすべての権利を付与するので、マルチプロセッシング・システムにアップグレードする際に、これらのライセンスを再インストールする必要はありません。システムを新しいマルチプロセッシング・レベルにアップグレードする場合には、既存のライセンスに SMP Extension License を追加してください。

13.2.2 マルチプロセッシング環境の監視

オペレーティング・システムのいくつかの機能により、マルチプロセッサ・システムの特性、能力、および状態に関する特別な情報を入手することができます。これらを可能にするものには、DCL の SHOW CPU コマンド、MONITOR ユーティリティなどがあります。

マルチプロセッサ構成に関する情報の入手

SHOW CPU コマンドにより、マルチプロセッシング・システムの構成および状態を示す 3 つのレベルの情報が表示されます。

レベル	コマンドの例	表示内容
要約	SHOW CPU	1 次プロセッサ、システム構成に加えられているプロセッサ、アクティブなプロセッサを示し、システム内のプロセッサの最低リビジョン・レベルとシステム・パラメータ MULTIPROCESSING の設定内容を表示する。また、マルチプロセッシング環境が使用可能になっているかどうかが表示される。
簡易	SHOW CPU/BRIEF	要約レベルの情報から抽出される。構成に追加されている各プロセッサの現在の CPU の状態と、現在稼働中のプロセスがあればその状態を示す。
詳細	SHOW CPU/FULL	要約レベルおよび追加の情報から抽出される。構成に追加されている各プロセッサの現在の CPU の状態、現在稼働中のプロセスがあればその状態、リビジョン・レベル、および能力を示す。また、特定のプロセッサ上でしか実行できないプロセスを示す。

SMP に関連する DCL のコマンドについては、『OpenVMS DCL ディクショナリ』を参照してください。MONITOR ユーティリティについては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の MONITOR のパートを参照してください。

13.3 ベクタ処理について

1 つの値を持つ単一のデータ項目を **スカラ** と呼びます。また、データ型が同じで互いに関連するスカラ値のグループを **ベクタ** と呼びます。

伝統的なスカラ・コンピュータはスカラ値に対してのみ動作するため、ベクタを扱う場合はその要素を順次処理する必要があります。一方、ベクタ・コンピュータはベクタをそのままのデータ構造として認識し、1 つのベクタ命令によってベクタ全体を操作することができます。複数の算術演算または論理演算が同時に行われるため、ベクタ・コンピュータで 1 つのベクタを処理すると、伝統的なコンピュータでスカラ命令だけを使用して同じベクタを処理する場合よりも 4 ~ 5 倍高速になります。

さらに、ベクタ・プロセッサはデータのストリームを高速に処理するための特殊なハードウェア技術によって高速な処理を実現しています。これらの技術として、データのパイプライン処理やチェイニングのほか、メモリ、算術機能ユニット、論理機能ユニットなどのハードウェアをさまざまな形式で並列化したことなどが挙げられます。ユニットをパイプライン化することによって、同じ処理フローの前後の演算を並行して行うことが可能になっています。

13.3.1 VAX におけるベクタ処理のサポート (VAX のみ)

VAX ベクタ・アーキテクチャには、それぞれが 64 個の要素を含む 16 個の 64 ビット・ベクタ・レジスタ (V0 ~ V15)、ベクタ・カウント・レジスタ (VCR) やベクタ長レジスタ (VLR)、ベクタ・マスク・レジスタ (VMR) などのベクタ制御レジスタ、ベクタ機能ユニット、ベクタ命令のセットが含まれています。VAX ベクタ命令は、ベクタ・レジスタとメモリとの間のデータ転送、整数および浮動小数点数の算術演算、およびプロセッサの制御を行います。VAX ベクタ・アーキテクチャ、ベクタ・レジスタ、およびベクタ命令についての詳細は、『VAX MACRO and Instruction Set Reference Manual』を参照してください。

VAX ベクタ・アーキテクチャに準拠した VAX システムを **ベクタ対応システム** と呼びます。

VAX ベクタ処理システムの構成には、スカラ CPU とベクタ CPU とを統合したプロセッサ (**ベクタ機能付きプロセッサ**) が 1 組以上含まれています。このようなシステム構成には、各スカラ・プロセッサに対して 1 つのベクタ・コプロセッサを持つ対称型構成、および付加的なスカラ専用プロセッサを組み込んだ非対称型構成があります。VAX ベクタ処理システムのモデルにより、ベクタ機能付きプロセッサは、スカラ CPU コンポーネントとベクタ CPU コンポーネントを物理的に統合した単一のモジュールから構成される場合と、それぞれのコンポーネント

を物理的に分離した 2 つのモジュールから構成される場合があります。どちらの場合も、スカラ CPU およびベクタ CPU は論理的に統合され、同じメモリを共用し、高速の専用内部パスを通じてデータを送信します。このように CPU が密接に結合しているため、ベクタ CPU を使用した場合は入出力動作が軽減されます。

VAX スカラ処理システムと同様、VAX ベクタ処理システムは VAXcluster のメンバ、ネットワーク上のノード、またはスタンドアロン・システムとして稼働させることができます。

13.3.2 VAX ベクタ命令エミュレーション機能 (VVIEF) (VAX のみ)

VAX ベクタ命令エミュレーション機能 (VVIEF) は、オペレーティング・システムに標準で備えられている機能であり、ベクタ・プロセッサが利用できない VAX システム上でベクタ化されたアプリケーションの作成やデバッグを可能にします。VVIEF は、非特権 VAX ベクタ命令とベクタ・システム・サービスを含む VAX ベクタ処理環境をエミュレートします。VVIEF の使用は、ユーザ・モードのコードに制限されています。

VVIEF は厳密にプログラム開発ツールであり、ベクタ・ハードウェア上での動作をソフトウェア上で実現するものではありません。したがって、VVIEF のもとで実行するようにアプリケーションをベクタ化しても性能は向上しません。むしろ、VVIEF のもとで実行されるベクタ化アプリケーションの動作はスカラ版アプリケーションよりも低速です。

オペレーティング・システムは、VVIEF ブートストラップ・コードをエグゼクティブ・ロードブル・イメージとして提供します。ここで注意が必要なのは、OpenVMS ベクタ・サポート・コードが存在すると、VVIEF は非稼働状態のままであるという点です。ベクタ機能付きシステムでベクタ・サポート・コードをロードしないようにして VVIEF を稼働させることは可能ですが、それによる利点はほとんどありません。ベクタ・サポート・コードのロードについては、13.4.1 項「ベクタ処理サポート・コードのロード (VAX のみ)」で説明します。

VVIEF のロードとアンロードについては、13.4.6 項「VAX ベクタ命令エミュレーション機能 (VVIEF) のロード (VAX のみ)」で説明します。

13.4 ベクタ処理環境の管理 (VAX のみ)

この節では、ベクタ処理システムの管理について取り上げます。

13.4.1 ベクタ処理サポート・コードのロード (VAX のみ)

VAX ベクタ処理システムでは、ブート時にベクタ処理サポート・コードが自動的にロードされるように設定されています。この省略時の動作は、静的なシステム・パラメータ VECTOR_PROC を設定することで変更することができます。具体的な設定方法は、表 13-1 「システム・パラメータ VECTOR_PROC の設定」に示すとおりです。

表 13-1 システム・パラメータ VECTOR_PROC の設定

値	結果
0	システム構成に関係なく、ベクタ処理サポート・コードをロードしない。
1	少なくとも 1 つのベクタ機能付きプロセッサが存在する場合にベクタ処理サポート・コードをロードする (省略時の設定)。
2	システムがベクタ対応の場合にベクタ処理サポート・コードをロードする。この設定は、各プロセッサごとに電源を供給しているシステムで最も有効。また、この設定では、オペレーティング・システムをリポートしなくてもベクタ・プロセッサをシステム構成に追加することができる。

13.4.2 ベクタ処理システムの構成 (VAX のみ)

マルチプロセッシング構成へのベクタ機能付きプロセッサの追加または削除は、システム・パラメータ SMP_CPUS を設定することによってブート時に行うことができます。また、DCL の START/CPU コマンドおよび STOP/CPU コマンドによって実行時に行うこともできます。オペレーティング・システムはベクタ機能付きプロセッサのスカラ CPU コンポーネントとベクタ

CPU コンポーネントとを単一のプロセッサとして扱い、これらの起動と停止を同時に行います。

ブート時、システム・パラメータ `SMP_CPUS` の設定に従って、マルチプロセッシング・システムの中の 2 次プロセッサのうちどのプロセッサを構成に追加するかが決定されます。これらの 2 次プロセッサの中にはベクタ機能付きプロセッサも含まれます。オペレーティング・システムは、1 次プロセッサを常に構成に追加します。システム・パラメータ `SMP_CPUS` の省略時の値 `-1` では、利用可能なすべてのプロセッサがブートされ、スカラ・プロセッサもベクタ機能付きプロセッサも同様にシステム構成に追加されます。この `SMP_CPUS` パラメータについての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照してください。ブート時、ベクタ機能付きプロセッサの起動の前に、ベクタ処理サポート・コード (13.4.1 項「ベクタ処理サポート・コードのロード (VAX のみ)」を参照) をロードしておく必要があります。サポート・コードがロードされていないければ、プロセスはベクタ機能付きプロセッサのスカラ CPU コンポーネントしか使用できなくなります。

稼働中のマルチプロセッシング・システムに 2 次プロセッサを追加する場合は、`DCL` の `START/CPU` コマンドを使用します。また、システムから 2 次プロセッサを削除する場合は、`STOP/CPU` コマンドを使用します。この場合でも、ベクタ処理サポート・コードがブート時にロードされていないければ、起動したベクタ機能付きプロセッサのベクタ CPU コンポーネントを使用することはできません。

`STOP/CPU` コマンドを使用する場合、そのコマンドによって削除するベクタ機能付きプロセッサが現在動作中のベクタ使用プロセスにベクタ機能を提供する唯一のものである場合、コマンドは異常終了し、メッセージが生成されます。プロセッサを修理するためにプロセッサを構成から削除するような特別なケースでは、`STOP/CPU/OVERRIDE` コマンドを発行すれば、ベクタ機能を使用するプロセスが残っていてもプロセッサを強制的に停止させることができます。

`STOP/CPU/OVERRIDE` コマンドがベクタ機能付きプロセッサに対して発行された場合、またはベクタ機能付きプロセッサが異常終了した場合、オペレーティング・システムは残されたすべてのベクタ使用プロセスをベクタ機能付きプロセッサが構成に追加されるまで「CPU 機能待ち」状態 (`RSN$_CPUCAP`) にします。その後、他のプロセス (潜在的ベクタ使用プロセス) からベクタ命令が発行されると、要求された CPU が稼働していないことを示すメッセージ (`CPUNOTACT`) が返されます。

`START/CPU` コマンドと `STOP/CPU` コマンドについての詳細は、『OpenVMS DCL ディクショナリ』を参照してください。

13.4.3 ベクタ・プロセスの管理 (VAX のみ)

オペレーティング・システムのスケジューリング・アルゴリズムにより、ベクタ処理資源とスカラ処理資源とはベクタ使用プロセス、潜在的ベクタ使用プロセス、スカラ使用プロセスに自動的に分配されます。しかし、VAX ベクタ処理構成は、次の 2 つの重要な点が異なります。

- 構成が収容する必要がある動作の量。
- ベクタ処理に必要なものを提供するためにシステム構成の中で利用可能なベクタ機能付きプロセッサの数。

システム内に存在するベクタ使用プロセスの数がそれらにサービスを提供するスカラ機能付きプロセッサ (スカラ・プロセッサとベクタ・プロセッサのペア) の数よりも多い場合は、プロセスの優先順位に従ってベクタ使用プロセスがベクタ機能付きプロセッサを共用します。ベクタ機能付きプロセッサを使用する複数のプロセスの優先順位が等しい場合、それらのプロセスはラウンド・ロビン形式でスケジューリングされます。ベクタ機能付きプロセッサに新しいベクタ使用プロセスをスケジューリングする場合、システムは現在メモリ内に存在するベクタ使用プロセスのベクタ・コンテキストを保存し、メモリから新しいベクタ使用プロセスを復元する必要があります。このような「低速」ベクタ・コンテキスト・スイッチが過剰に発生する場合は、実際の演算よりもベクタ・コンテキスト・スイッチに多くの処理時間が費やされるようになります。

ベクタ処理の必要性が大きなシステムは、それらの必要性を満たすように構成する必要があります。しかし、既存の構成のままチューニングを行うことでその性能を改善できるいくつかの方法があります。

13.4.3.1 システム資源とプロセス・クォータの調整 (VAX のみ)

いくつかのベクタ使用プロセスが同時に稼働状態になるシステムでは、プロセスがメモリを共用するためにページング動作が増加することがあります。プロセス・ページングの回数を減らすためには、AUTHORIZE ユーティリティを使って、ベクタ化アプリケーションを実行しているプロセスのワーキング・セットの制限とクォータを調整します (詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』の AUTHORIZE のパートを参照してください)。また、プロセス最大ワーキング・セット・サイズ (システム・パラメータ WSMAX) の値を大きくします。さらに必要であれば、ベクタ化アプリケーションでワーキング・セット・システム・サービス (\$LKWSET) のロック・ページを使って、自身の性能を向上させます。

ベクタを使用する各プロセスは 8 KB のシステム非ページング動的メモリを割り当てられ、その中にベクタ・コンテキストの情報が格納されます。システム内で同時に稼働状態になっているベクタ使用プロセスの数に応じて、システム・パラメータ NPAGEDYN の値を調整します。DCL の SHOW MEMORY/POOL/FULL コマンドを使用すると、非ページング・プールの現在のサイズがバイト数で示されます。

VAX ベクタ処理システムの最適な性能を得るため、バッチ・キューの設定時にシステムのベクタ資源を飽和させないように注意する必要があります。キューに登録された稼働状態のベクタ化バッチ・ジョブの数がシステム内のベクタ機能付きプロセッサの数より多いと、ベクタ・コンテキスト・スイッチに処理時間の多くの部分が消費されるようになります。

できれば、ベクタ化されたバッチ・ジョブを VAX ベクタ処理システムにディスパッチするためのキュー (たとえば VECTOR_BATCH) を別に用意し、そのジョブの制限数をシステム内のベクタ機能付きプロセッサの数と同じにします。そして、ベクタ化バッチ・ジョブをこの汎用ベクタ処理バッチ・キューに登録するようにユーザに指示します。

13.4.3.2 プロセスへのスカラ資源およびベクタ資源の配分 (VAX のみ)

ベクタ使用プロセスはベクタ機能付きプロセッサにスケジューリングする必要があります。プロセスが実行しているイメージが、ある期間スカラ命令だけを発行し、ベクタ機能付きプロセッサを他のベクタ使用プロセスと共用する必要がある場合、そのイメージはスカラ・プロセッサ上で実行できないために、そのイメージ自身の性能だけでなく、システム全体の性能が低下します。

省略時の設定では、ベクタ使用プロセスがある期間ベクタ命令を発行していない場合、そのプロセスはすぐにはベクタ命令を発行しないと想定されます。このプロセスはベクタ機能が必要でないといみなされ、潜在的ベクタ使用プロセスに移行します。

非対称型ベクタ処理構成では、潜在的ベクタ使用プロセスが検出されると次のような効果があります。

- 潜在的ベクタ使用プロセスはより大きなセット (スカラ・プロセッサとベクタ機能付きプロセッサの両方) で実行できるため、その応答時間が向上する。
- 潜在的ベクタ使用プロセスをスカラ・プロセッサにスケジューリングすることにより、ベクタ機能付きプロセッサの競合が少なくなる。
- ベクタ命令を発行するベクタ使用プロセスがベクタ機能付きプロセッサにスケジューリングされるようになり、ベクタ CPU がより効果的に使用される。

システム・パラメータ VECTOR_MARGIN は、すべてのベクタ使用プロセスの状態がチェックされる間隔を決定します。VECTOR_MARGIN に、1 から FFFFFFFF₁₆ までの任意の整数値に設定することができます。この値は、システム・パラメータ QUANTUM によって決定される連続プロセス・クォンタムの数を表します。指定したクォンタムの数で決定する期間プロセスがベクタ命令を発行しないと、システムはそのプロセスを潜在的ベクタ使用プロセスとして宣言します。

VECTOR_MARGIN の省略時の値は 200₁₀ です。

13.4.4 ACL によるベクタ・プロセッサへのアクセスの制限 (VAX のみ)

ベクタ機能は、ユーザがベクタ・プロセッサのサービスを使用できるようにするために、オペレーティング・システムが使用する抽象的なソフトウェアです。アクセス制御リスト (ACL) に

ベクタ機能のオブジェクトを対応づけることによって、ベクタ・プロセッサの使用を特定のIDを持つユーザに制限することができます。

たとえば、大学ではベクタ・プロセッサの使用を、イメージ処理を研究する学部や学生に限定する場合があります。また、情報サービス会社では、ベクタ機能へのアクセスとベクタ・プロセッサを使用した時間の一方または両方をユーザに対して請求する場合があります。

DCL コマンド SET SECURITY/ACL を次の形式で使用して、ベクタ機能についての ACL を設定します。

```
SET SECURITY /CLASS=CAPABILITY /ACL=(ace[,...]) VECTOR
```

次の DCL コマンドでベクタ機能のが表示されます。

```
§ SHOW SECURITY /CLASS=CAPABILITY VECTOR
```

ここで、ACLはベクタ機能に対して設定されるものであり、システム上の任意またはすべてのベクタ機能付きプロセッサの使用を制限するものではないということに注意してください。オペレーティング・システムは、ベクタ機能付きプロセッサ上のベクタ機能を使用する許可を持たないプロセスでもスケジューリングします。しかし、これらのプロセスが利用できるのはプロセッサのスカラ CPU コンポーネントだけであって、ベクタ命令を実行することはできません。同様に、ACLはベクタ機能に対して設定され、ベクタ機能付きプロセッサに対して設定されているものでないため、ACLを使って特定のプロセッサ上でジョブを長時間実行させることはできません。

SET SECURITY コマンドと SHOW SECURITY コマンドについての詳細は、『OpenVMS DCL ディクショナリ』を参照してください。

13.4.5 ベクタ処理システムに関する情報の入手 (VAX のみ)

ベクタ処理システムの状態とプロセス別のシステム使用量についての情報をさまざまな方法で入手することができます。具体的には、次のとおりです。

- DCL レキシカル関数 F\$GETJPI および F\$GETSYI
- DCL の SHOW CPU コマンド
- DCL のコマンド SHOW PROCESS および LOGOUT/FULL
- ACCOUNTING ユーティリティ
- MONITOR ユーティリティ

13.4.5.1 DCL レキシカル関数 F\$GETJPI および F\$GETSYI (VAX のみ)

DCL レキシカル関数 F\$GETJPI は以下の項目を受け付け、指定されたプロセスのベクタ状態に関して対応する情報を返します。

項目	戻り値の型	返される情報
FAST_VP_SWITCH	整数	指定されたプロセスが発行したベクタ命令のうち、ベクタ・コンテキスト・スイッチを行うことなく非稼働状態のベクタ・プロセッサを稼働状態にした命令の数。
SLOW_VP_SWITCH	整数	指定されたプロセスが発行したベクタ命令のうち、完全なベクタ・コンテキスト・スイッチを行うことによって非稼働状態のベクタ・プロセッサを稼働状態にした命令の数。
VP_CONSUMER	論理	指定されたプロセスがベクタ使用プロセスかどうかを示すフラグ。
VP_CPUTIM	整数	指定されたプロセスがベクタ機能を利用した時間の累計。

DCL レキシカル関数 F\$GETSYI は以下の項目を受け付け、ベクタ処理システムの状態に関して対応する情報を返します。

項目	戻り値の型	返される情報
VECTOR_EMULATOR	整数	システムに VAX ベクタ命令エミュレーション機能 (VVIEF) が存在することを示すフラグ。
VP_MASK	整数	システム内のどのプロセッサがベクタ・コプロセッサを持っているかを示すマスク。
VP_NUMBER	整数	システム内のベクタ・プロセッサの数。

DCL レキシカル関数の F\$GETJPI および F\$GETSYI についての詳細は、『OpenVMS DCL ディクショナリ』を参照してください。

13.4.5.2 SHOW CPU/FULL コマンド (VAX のみ)

SHOW CPU/FULL コマンドにより、指定された CPU の機能の一覧が示されます。STOP/CPU コマンドを実行する場合、まずこのコマンドを実行してシステムがベクタ機能を持っているかどうかを調べてください。

SHOW CPU コマンドについての詳細は、『OpenVMS DCL ディクショナリ』を参照してください。

13.4.5.3 SHOW PROCESS コマンドと LOGOUT/FULL コマンド (VAX のみ)

あるプロセスがベクタ機能付きプロセッサにスケジューリングされたベクタ使用プロセスとして任意の時間が経過すると、DCL の SHOW PROCESS コマンドと LOGOUT/FULL コマンドによって、そのプロセスがベクタ CPU を実際に使用した時間とそのプロセスにベクタ CPU が割り当てられた時間がそれぞれ表示されます。

ベクタ CPU 時間を累計するためには、プロセスがベクタ使用プロセス (すなわち、システム・ベクタ機能を必要とするプロセス) であり、ベクタ機能付きプロセッサにスケジューリングされていることが必要です。オペレーティング・システムは、ベクタ使用プロセスをベクタ機能付きプロセッサにスケジューリングすると、そのプロセスが実際にはベクタ CPU を使用していなくてもそのプロセスのベクタ CPU 時間を累計します。スカラ使用プロセスと潜在的ベクタ使用プロセスはベクタ CPU を使用しないため、これらのプロセスがベクタ機能付きプロセッサにスケジューリングされているときでも、それらのベクタ CPU 時間は累計されません。

SHOW PROCESS コマンドおよび LOGOUT コマンドについての詳細は、『OpenVMS DCL ディクショナリ』を参照してください。

13.4.6 VAX ベクタ命令エミュレーション機能 (VVIEF) のロード (VAX のみ)

VAX ベクタ命令エミュレーション機能 (VVIEF) は、オペレーティング・システムに標準で備えられている機能であり、ベクタ・プロセッサが利用できない VAX システム上でベクタ化されたアプリケーションの作成やデバッグを可能にします。VVIEF は厳密にプログラム開発ツールとして意図されたものであり、ベクタ・ハードウェア上での動作をソフトウェア上で実現するものではありません。したがって、VVIEF のもとで実行するようにアプリケーションをベクタ化しても性能は向上しません。むしろ、VVIEF のもとで実行されるベクタ化アプリケーションの動作はスカラ版アプリケーションよりも低速です。

コマンド・プロシージャ SYS\$UPDATE:VVIEF\$INSTAL.COM を呼び出すと、次回以降のシステム・ブート時に VVIEF がロードされます。VVIEF をアンロードするためには、コマンド・プロシージャ SYS\$UPDATE:VVIEF\$DEINSTAL.COM を呼び出して、システムをリポートします。

システムに VVIEF が存在するかどうかは DCL の以下のコマンドによって調べることができます。

```
$ X = F$GETSYI ("VECTOR_EMULATOR")
```

```
$ SHOW SYMBOL X
```

```
X = 1   Hex = 00000001   Octal = 0000000001
```

戻り値が 1 の場合は VVIEF が存在し、0 の場合は存在しません。

ベクタ・サポート・コードが存在しても VVIEF をシステムにロードすることはできませんが、その場合は非稼働状態のままになります。ベクタ機能付きシステム (13.4.1 項「ベクタ処理サポート・コードのロード (VAX のみ)」を参照) でベクタ・サポート・コードをロードしないようにして VVIEF を稼働させることは可能ですが、それによる利点はほとんどありません。システム内でベクタ機能付きプロセッサだけに障害が発生した場合、中断されたベクタ化されたアプリケーションの実行は VVIEF のもとでは再開されません。

第14章 DECdtm サービスの管理

この章では、DECdtm サービスを利用するソフトウェア (たとえば ACMS、Oracle Rdb、RMS Journaling など) を実行するときに必要な作業を説明します。



注意:

OpenVMS Alpha システムや I64 システムでは、DECdtm サービスがマルチスレッド環境で使用されると、予測できない結果が生じる場合があります。DECdtm が実行する処理の多くでは、呼び出しプロセスのコンテキストを使用するため、初期スレッド以外のカーネルスレッド内の DECdtm サービスを呼び出さないでください。

この章の内容

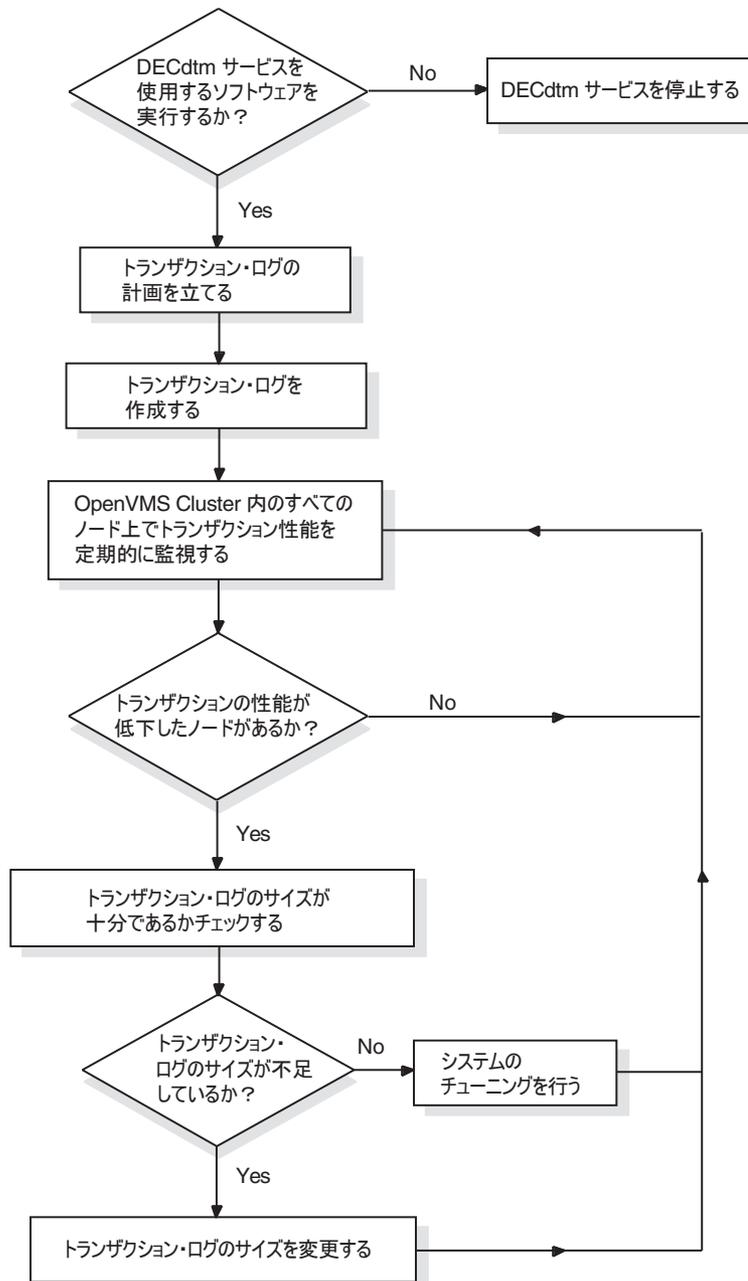
この章では次の作業について説明します。

作業	参照箇所
トランザクション・ログの計画	14.2 項 「トランザクション・ログの計画」
DECnet-Plus ネットワークの計画	14.3 項 「DECnet-Plus ネットワークの計画」
トランザクション・ログの作成	14.4 項 「トランザクション・ログの作成」
トランザクション性能の監視	14.5 項 「トランザクション性能の監視」
トランザクション・ログのサイズが十分かどうかのチェック	14.6 項 「トランザクション・ログのサイズが十分かどうかのチェック」
トランザクション・ログのサイズの変更	14.7 項 「トランザクション・ログのサイズの変更」
トランザクション・ログの移動	14.8 項 「トランザクション・ログの移動」
ディスクのディスマウント	14.9 項 「ディスクのディスマウント」
ノードの追加	14.10 項 「ノードの追加」
ノードの削除	14.11 項 「ノードの削除」
DECdtm サービスの停止	14.12 項 「DECdtm サービスの停止」
DECdtm サービスの開始	14.13 項 「DECdtm サービスの開始」
XA Gateway の使用 (Alpha および I64)	14.14 項 「XA Gateway の使用 (Alpha および I64)」

図 14-1 「DECdtm サービスの管理」は、DECdtm サービスの管理のためのフローチャートです。このチャートから実際に必要な作業と、その作業の順序を知ることができます。さらに、次の項目について説明します。

項目	参照箇所
トランザクション・ログについて	14.1 項 「トランザクション・ログについて」
トランザクション・グループについて	14.3.2.2 項 「トランザクション・グループについて」

図 14-1 DECdtm サービスの管理



ZK5154AGE

14.1 トランザクション・ログについて

トランザクション・ログとは、あるノード上で行われた DECdtm トランザクションに関する情報を格納するファイルのことです。ファイルのタイプは .LM\$JOURNAL です。

任意のノードで DECdtm トランザクションを実行する場合は、そのノードに対してトランザクション・ログを作成しておく必要があります。OpenVMS Cluster では、クラスタ内の各ノードでトランザクション・ログを作成します。ログ・マネージャ制御プログラム (LMCP) ユーティリティを使用して、トランザクション・ログの作成と管理を行います。

DECdtm サービスでは、トランザクション・ログの格納場所は論理名 SYS\$JOURNAL によって判断されます。トランザクション・ログを含むディレクトリを指すように、SYS\$JOURNAL に定義する必要があります。

14.2 トランザクション・ログの計画

トランザクション・ログのサイズと格納場所は、トランザクションの性能に影響を及ぼします。トランザクション・ログを作成する前に、そのサイズと格納場所を決定してください。

トランザクション・ログのサイズと格納場所は、作成した後でも変更することができます。しかし、この段階でログの格納場所とサイズについて十分に検討しておく、後になって変更する負担が少なくて済みます。

この節では次の作業について説明します。

作業	参照箇所
トランザクション・ログのサイズの決定	14.2.1 項 「トランザクション・ログのサイズの決定」
トランザクション・ログの格納場所の決定	14.2.2 項 「トランザクション・ログの格納場所の決定」

14.2.1 トランザクション・ログのサイズの決定

トランザクション・ログを作成する場合は、そのサイズをブロック単位で指定することができます。省略時のサイズは 4,000 ブロックです。この値に設定すれば、ほとんどのシステムで十分な性能を得ることができます。

トランザクションの平均発生数が予想できる場合、トランザクション・ログのサイズは以下の計算式で求めることができます。

$$\text{サイズ} = 40 \times \text{平均発生数}$$

サイズ	トランザクション・ログのサイズを示すブロック数
平均発生率	1 秒あたりに実行されるトランザクション数の平均

トランザクションの平均発生数が分からない場合には、省略時の値の 4,000 ブロックを採用します。

14.2.2 トランザクション・ログの格納場所の決定

できれば、以下の属性を持つディスクに置いてください。

高速	半導体ディスクのような高性能ディスクで、負荷の低いもの。
高可用性	データに複数のアクセス・パスを設定することによって高い可用性を実現することができます。 OpenVMS Cluster では、クラスタ内の別のノードからもアクセスできるディスクを使用する。1つのノードがダウンした場合にも、別のノードで実行されているトランザクションはブロックされない。
高信頼性	データの複数のコピーを作成しておけば、高い信頼性を実現することができる。 シャドウ・ディスクは非シャドウ・ディスクよりも信頼性が高いが、トランザクション・ログがほぼ書き込み専用である分低速になる。

速度と可用性または信頼性のいずれかとの間で選択を行う必要があることがあります。たとえば、ノードがワークステーションの場合には、可用性と信頼性のために速度を犠牲にして、ワークステーションに接続されたディスクではなく、それより低速のHSCベースのシャドウ・ディスク上にノードのトランザクション・ログを置く場合があります。

クラスタ環境では、できるだけトランザクション・ログを複数のディスクに分散させてください。1つのディスクに複数のトランザクション・ログが存在すると、トランザクションの性能が低下します。



注意:

ディスクにトランザクション・ログを保持するための十分な連続領域があることを確認してください。トランザクション・ログの領域が連続していないと、トランザクション性能は低下します。

14.3 DECnet-Plus ネットワークの計画

この節では、DECnet-Plus ネットワークで DECdtm を使用する際に役立つ、次の情報についてまとめます。

- DECnet-Plus ネームスペースの計画
- DCEnet-Plus ネットワークでの SCSNODE 名の計画

14.3.1 DECnet-Plus ネームスペースの計画

DECdtm では、複数の DECnet-Plus ネームスペースをサポートしません。

つまり、DECdtm サービスを使用するソフトウェアを使用したい場合には、ローカル・ネームスペースと DECdns ネームスペースの両方を使用することはできません。

14.3.2 DCEnet-Plus ネットワークでの SCSNODE 名の計画

SCSNODE は、コンピュータの名前を定義するシステム・パラメータです。DECnet-Plus ネットワークを使用しているときに、異なる OpenVMS Cluster または異なるスタンドアロン・コンピュータにまたがる DECdtm トランザクションを実行したい場合には、SCSNODE 名を選択する際に次の規則に従う必要があります。

14.3.2.1 SCSNODE 名に関する規則

DECnet-Plus ネットワークを使用していて、異なる OpenVMS Cluster または異なるスタンドアロン・コンピュータにまたがる DECdtm トランザクションを実行する場合には、SCSNODE 名が次の規則に従っているかどうかを確認する必要があります。

- トランザクション・グループ内の各コンピュータの SCSNODE 名は、次の名前と異なる名前であればならない。
 - トランザクション・グループ内の他のコンピュータの SCSNODE 名。SCSNODE 名は、同じトランザクション・グループ内で一意な名前であればならない。
 - 同じローカル・ルートの他のコンピュータの DECnet 単純名。
 - ネットワーク全体の他のコンピュータの DECnet 同意語。トランザクション・グループについての詳細は、14.3.2.2 項「トランザクション・グループについて」を参照。
- コンピュータが OpenVMS Cluster 内にあるときは、SCSNODE 名は次の名前と異なる名前であればならない。
 - 同じクラスタ内の他のコンピュータの DECnet 単純名。
 - 他のクラスタ・メンバと同じローカル・ルートのコンピュータの DECnet 単純名。



注記: HP DECnet-Plus for OpenVMS ネットワーク接続を持ち、IP ルータだけを使用してノードが接続されている遠隔システムとの分散トランザクションに参加するためにユーザ作成のアプリケーションが DECdtm を呼び出している場合、このアプリケーションを実行した ACMS ユーザ、Rdb ユーザ、およびその他のユーザには、次のエラーが DECnet から返されます。

IPC-E-BCKTRNSFAIL, failure on the back translate address request

このエラーは、DECnet-Plus が遠隔ノードの名前を変換できないという論理接続の障害時に表示されます。このエラーは、次の場合に発生する可能性があります。

1. 遠隔システムの DECnet-Plus ノード名が、ローカルの DECnet-Plus データベースには定義されておらず、遠隔ノードの TCP/IP ネーム・サーバの ALIAS としてだけ定義されている。たとえば、ノード XXYZZY が、次のように定義されている場合。

20.43.136.54 XXYZZY.ABC.DEF.COM, XXYZZY

2. ノード名が、DECnet データベースで解決できない。この場合、アドレス解決メカニズムは、TCP/IP データベースを利用する。それでもノード名が解決できない場合は、サービスは上記のエラーで失敗する。

この状況を回避するには、TCP/IP データベースに SCSNAME を登録する。

14.3.2.2 トランザクション・グループについて

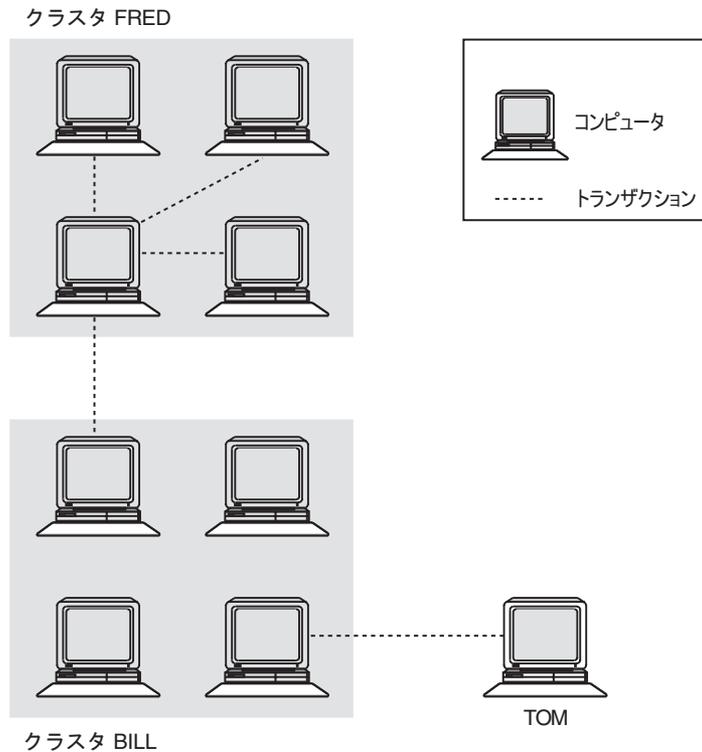
トランザクション・グループとは、DECdtm トランザクションに関するコンピュータのグループであり、その SCSNODE 名は、14.3.2.1 項「SCSNODE 名に関する規則」で説明する規則に従わなければなりません。

トランザクション・グループは、次のガイドラインに従います。

- 各コンピュータは 1 つのトランザクション・グループに属す。
- OpenVMS Cluster 内のすべてのコンピュータは、同一トランザクション・グループに属す。
- 1 つのトランザクションがコンピュータ A とコンピュータ B にまたがる場合には、コンピュータ A とコンピュータ B は同一トランザクション・グループに属す。

図 14-2 「トランザクション・グループ」は、トランザクション・グループの例を示していません。

図 14-2 トランザクション・グループ



ZK6302AG E

この図で、9台のコンピュータはすべて同じトランザクション・グループに属します。これは次の理由によります。

- トランザクションは、クラスタ FRED 内のコンピュータとクラスタ BILL 内のコンピュータにまたがっている。つまり、クラスタ FRED 内の4台のコンピュータと、クラスタ BILL 内の4台のコンピュータは、同じトランザクション・グループに属す。
- トランザクションは、スタンドアロン・コンピュータ TOM とクラスタ BILL 内のコンピュータにまたがっている。つまり、スタンドアロン・コンピュータ TOM は、クラスタ BILL 内のコンピュータと同じトランザクション・グループに属す。

14.4 トランザクション・ログの作成

任意のノードで DECdtm サービスを利用する場合は、そのノードに対してトランザクション・ログを作成する必要があります。OpenVMS Cluster 環境では、ノードごとにトランザクション・ログを作成します。



重要:

トランザクション・ログの作成後にクラスタからノードを削除すると、データが破壊されることがあります。ノードを安全に削除する方法については、14.11 項「ノードの削除」を参照してください。

作業方法

1. 14.2 項「トランザクション・ログの計画」のガイドラインに従って、ノードごとにトランザクション・ログのサイズと格納場所を決定する。ディスクにはトランザクション・ログを格納するのに十分な連続領域が存在する必要がある。
2. クラスタ環境の場合、トランザクション・ログを作成するディスクがクラスタ全体でマウントされていることを確認する。

3. トランザクション・ログを作成するディレクトリを決定する。必要であれば、トランザクション・ログ専用のディレクトリを作成する。
4. 次のようにトランザクション・ログのディレクトリを SYS\$JOURNAL に定義する。
DO DEFINE/SYSTEM/EXECUTIVE_MODE SYS\$JOURNAL ディレクトリ指定 [...]
ここで、ディレクトリ指定は、作成する 1 つ以上のトランザクション・ログを格納するディレクトリの完全ファイル指定である。このとき、トランザクション・ログを格納するすべてのディレクトリを指定する必要がある。指定する順序は自由。
クラスタ環境では、SYSMAN を使用して SYS\$JOURNAL をクラスタ全体で定義する。
5. コマンド・プロシージャ SYS\$MANAGER:SYLOGICALS.COM に、SYS\$JOURNAL の定義を追加する。
独自の SYLOGICALS.COM を使用しているすべてのノードで同じ作業を行う。
6. LMCP の CREATE LOG コマンドを使用して、ノードごとに 1 つのトランザクション・ログを作成する。
CREATE LOG [/SIZE= サイズ] ディレクトリ指定 SYSTEM\$ ノード.LM\$JOURNAL

サイズ トランザクション・ログのサイズを示すブロック数。サイズの指定を省略すると、トランザクション・ログのサイズは 4,000 ブロックになる。

ディレクトリ指定 トランザクション・ログを格納するディレクトリの完全指定。

ノード ノードの名前。

7. DECdtm サービスを次のように開始する。

手順	作業
a.	論理名 SYS\$DECDTM_INHIBIT が定義されているかどうかを調べる。 \$ SHOW LOGICAL SYS\$DECDTM_INHIBIT
b.	SYS\$DECDTM_INHIBIT の定義状態 定義済み DECdtm サービスは停止される。14.13 項「DECdtm サービスの開始」の指示に従って DECdtm サービスを開始する。 未定義 DECdtm サービスが開始される。

例

この例では、OpenVMS Cluster 内のノード (SCSNODE 名が BLUE および RED) にトランザクション・ログを作成する方法を示します。どちらのノードも、ノード固有の SYLOGICALS.COM を使用していないものとします。

トランザクション・ログの格納場所とサイズを次のように決定します。

ノード	ログのサイズ (ブロック数)	ディスク
BLUE	5000	DUA1
RED	4000	DUA2

ディスクをクラスタ全体でマウントします。

```
$ MOUNT/CLUSTER/SYSTEM DUA1: LOG1
```

```
$ MOUNT/CLUSTER/SYSTEM DUA2: LOG2
```

トランザクション・ログ用のディレクトリを作成します。

```
$ CREATE/DIRECTORY DISK$LOG1:[LOGFILES]
$ CREATE/DIRECTORY DISK$LOG2:[LOGFILES]
```

SYS\$JOURNAL を定義します。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> SET ENVIRONMENT/CLUSTER
SYSMAN> DO DEFINE/SYSTEM/EXECUTIVE_MODE SYS$JOURNAL -
SYSMAN> DISK$LOG1:[LOGFILES], DISK$LOG2:[LOGFILES]
SYSMAN> EXIT
```

コマンド・プロシージャ SYS\$MANAGER:SYLOGICALS.COM に以下の行を追加します。

```
$ !
$ DEFINE/SYSTEM/EXECUTIVE_MODE SYS$JOURNAL DISK$LOG1:[LOGFILES], -
SYS$DISK$LOG2:[LOGFILES]
$ !
```

トランザクション・ログを作成します。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> CREATE LOG/SIZE=5000 DISK$LOG1:[LOGFILES] SYSTEM$BLUE.LM$JOURNAL
LMCP> CREATE LOG DISK$LOG2:[LOGFILES] SYSTEM$RED.LM$JOURNAL
LMCP> EXIT
```

DECdtm サービスが開始されたことを確認します。

```
$ SHOW LOGICAL SYS$DECDTM_INHIBIT
%SHOW-S-NOTRAN, no translation for logical name SYS$DECDTM_INHIBIT
SYS$DECDTM_INHIBIT が定義されていないため、DECdtm サービスが使用できます。
```

14.5 トランザクション性能の監視

負荷の増加など、システム構成を変更することでトランザクション性能に影響がでることがあります。毎月一度はノード上でトランザクションを監視し、トランザクション性能が低下していないことを確認してください。OpenVMS Cluster 環境では、クラスタ内のすべてのノードでトランザクション性能を監視してください。

作業方法

1. MONITOR ユーティリティの MONITOR TRANSACTIONS コマンドを使用してトランザクションを監視する。

```
MONITOR TRANSACTION/SUMMARY[= ファイル指定 ]
/ENDING= 終了時刻 /NODE= ノード名 [...]
```

ファイル指定 要約ファイルのファイル指定。トランザクションの情報は要約され、ここで指定したファイルに記録される。ファイル指定を省略すると、トランザクション情報は省略時のディレクトリの MONITOR.SUM に記録される。

終了時刻 監視セッションを終了する時刻。

ノード名 ノードの名前。OpenVMS Cluster では、クラスタ内のすべてのノードをリストする。

最適な情報を得るためには、トランザクションの監視を 24 時間連続して行うこと。

MONITOR TRANSACTION コマンドをコマンド・プロシージャに追加すれば、トランザクションの監視をバッチ・モードで行うことができる。

MONITOR TRANSACTION コマンドについての詳細は、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照。

2. 要約ファイルを調査する。

要約ファイルには、いくつかの異なるデータ項目の値が含まれている。各ノードについて、以下の項目に注目する。

- 1秒あたりに終了したトランザクション数の平均。
- 実行時間(発生から終了までの時間)別のトランザクション数の平均。
 - 1秒未満
 - 1秒～2秒
 - 2秒～3秒
 - 3秒～4秒
 - 4秒～5秒
 - 5秒以上

以上の値をメモしておく。

3. 今回の監視セッションの結果を前回のセッションの結果と比較する。

作業負荷が変化していなければ、トランザクションの発生数と実行時間はほとんど同じはずである。次のような場合は性能が低下していると考えられる。

- 1秒あたりに終了したトランザクション数の平均値が小さくなっている。
- トランザクションの実行時間の平均値が大きくなっている。

トランザクションの実行時間の平均値が大きくなっているかどうかを調べるには、実行時間別のトランザクションの平均数を比較する。実行時間が長くなったトランザクションが多ければ、実行時間の平均値が大きくなっていることになる。

いくつかの監視セッションを通じて各値がどのように変化したかを調べる。1つの監視セッションから次の監視セッションの間に見られる変化は、システム上の作業負荷の変化が原因であることが考えられる。

任意のノードでトランザクション性能が低下したと考えられる場合は、そのトランザクション・ログのサイズが十分かどうかチェックする(14.6項「トランザクション・ログのサイズが十分かどうかのチェック」を参照)。

トランザクション・ログのサイズが十分であるにもかかわらず、トランザクション性能が低下している場合は、システムのチューニングを検討する。システムのチューニングについての詳細は、『OpenVMS Performance Management』を参照。

例

この例では BLUE と RED の2つのノードを持つ OpenVMS Cluster のトランザクション性能を監視しています。

ノード BLUE と RED のトランザクションは 24 時間連続して監視します。

```
$ MONITOR TRANSACTION/SUMMARY=DISK$LOG1:[LOGFILES] TRANSACTIONS.SUM -  
_ $ /ENDING="+1-"/NODE=(BLUE,RED)
```

要約ファイルを調べます。

```
          DISTRIBUTED TRANSACTION STATISTICS  
          on node BLUE          From: 16-OCT-2000 14:23:51  
          SUMMARY              To:   17-OCT-2000 14:23:51  
          CUR          AVE          MIN          MAX  
Start Rate          49.02          43.21          31.30          49.02  
Prepare Rate        48.70          43.23          30.67          48.70  
One Phase Commit Rate 0.00          0.00          0.00          0.00  
Total Commit Rate   48.70          43.19          31.30          48.70  
Abort Rate          0.00          0.00          0.00          0.00  
End Rate            48.70          43.19          31.30          48.70
```

Remote Start Rate		0.00	0.00	0.00	0.00
Remote Add Rate		0.00	0.00	0.00	0.00
Completion Rate	0-1	21.42	13.57	0.63	21.42
by Duration	1-2	25.97	29.15	24.59	33.87
in Seconds	2-3	1.29	0.47	0.00	4.47
	3-4	0.00	0.00	0.00	0.00
	4-5	0.00	0.00	0.00	0.00
	5+	0.00	0.00	0.00	0.00

SUMMARIZING

DISTRIBUTED TRANSACTION STATISTICS

on node RED From: 16-OCT-2000 14:23:52
SUMMARY To: 17-OCT-2000 14:23:52

以下の値に注目します。

- 平均終了トランザクション数。
ノード BLUE の場合、1 秒あたり平均 43.19 のトランザクションが終了している。
- 実行時間別平均トランザクション数。
ノード BLUE の場合は以下のとおり。
 - 0 ~ 1 秒で終了したトランザクション数は 13.57
 - 1 ~ 2 秒で終了したトランザクション数は 29.15
 - 2 ~ 3 秒で終了したトランザクション数は 0.47

今回の監視セッションの結果を以前のセッションの結果と比較します。

セッション	終了数	実行時間別トランザクション数		
		0 ~ 1 秒	1 ~ 2 秒	2 ~ 3 秒
6 月	42.13	12.98	28.13	1.02
7 月	38.16	10.35	25.80	2.01
8 月	43.19	13.57	29.15	0.47

ノード BLUE のデータを見るかぎり、性能が低下した兆候は見られません。

14.6 トランザクション・ログのサイズが十分かどうかのチェック

トランザクション性能が低下しているノードが見つかった場合は、そのノードのトランザクション・ログのサイズが十分かどうかを調べる必要があります。

14.5 項「トランザクション性能の監視」では、トランザクション性能の低下を発見する方法を説明しています。

作業方法

1. トランザクション・ログが置かれているノードにログインする。
2. LMCP ユーティリティの SHOW LOG/CURRENT コマンドを使用して、トランザクション・ログがストールした回数を調べる。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> SHOW LOG/CURRENT
```

表示されたチェックポイントと発生したストールの数に注目する。

3. 5分後に SHOW LOG/CURRENT コマンドを繰り返し実行する。実行後、チェックポイントとストールの数に再び注目する。
4. SHOW LOG/CURRENT コマンドからの情報と比較する。
2回のチェックでチェックポイントの数が同じ場合は、システムの負荷が大きい時間に同じ作業をもう一度行う。
チェックポイントの数が増え、さらにストールの回数が1回以上増えている場合は、トランザクション・ログが小さすぎるといえる。
5. トランザクション・ログのサイズが小さいときは、そのサイズを大きくする。トランザクション・ログのサイズを変更する方法については、14.7 項「トランザクション・ログのサイズの変更」を参照。

例

ノード BLUE のトランザクション・ログのサイズが不十分かどうかを調べます。

ノード BLUE にログインし、トランザクション・ログがストールした回数を調べます。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> SHOW LOG/CURRENT
```

```
Checkpoint starts/ends          2464/2464
Stall starts/ends                21/21
Log status: no checkpoint in progress, no stall in progress.
```

チェックポイントの数は 2,464 で、トランザクションのストール回数は 21 回であることが分かります。

5分後、SHOW LOG/CURRENT コマンドをもう一度入力します。

```
LMCP> SHOW LOG/CURRENT
```

```
Checkpoint starts/ends          2514/2514
Stall starts/ends                28/28
Log status: no checkpoint in progress, no stall in progress.
```

チェックポイントの数が増え、さらにトランザクション・ログが現在 28 回ストールしており、ストール回数が 5 分間に 7 回増えていることが分かります。したがって、トランザクション・ログのサイズが不十分であるといえます。

14.7 トランザクション・ログのサイズの変更

トランザクション・ログのサイズが不十分な場合、そのサイズを大きくする必要があります。14.6 項「トランザクション・ログのサイズが十分かどうかのチェック」を参照してください。

作業方法



重要:

以下の手順で示すステップのすべてを実行してください。途中のステップを省略すると、データが壊れることがあります。

1. トランザクション・ログに対応するノードにログインする。
2. LMCP の SHOW LOG コマンドを使用して、トランザクション・ログが置かれているディレクトリを探す。
SHOW LOG SYSTEM\$ ノード.LM\$JOURNAL
ノードは、トランザクション・ログが置かれているノードの名前。
3. トランザクション・ログをリネームする。

RENAME ディレクトリ指定 SYSTEM\$ ノード .LM\$JOURNAL ディレクトリ指定 SYSTEM\$ ノード.LM\$OLD

ディレクトリ指定 トランザクション・ログを格納するディレクトリの完全指定。
ノード トランザクション・ログに対応するノードの名前。

4. ノードを全くシャットダウンしないで DECdtm サービスを使用しているすべてのソフトウェアを停止できるか。

可能な場合は、次のようにトランザクション・ログを閉じる。

手順	作業
a.	DECdtm サービスを使用しているすべてのソフトウェアを終了する。
b.	LMCP の CLOSE LOG コマンドを使用してトランザクション・ログを閉じる。 \$ RUN SYS\$SYSTEM:LMCP LMCP> CLOSE LOG CLOSE LOG コマンドはトランザクション・ログを閉じてから DECdtm TP_SERVER プロセスを停止する。DECdtm サービスを使用しているソフトウェアがあると、コマンドは失敗する。
c.	CLOSE LOG コマンドの実行に成功したか。 成功 TP_SERVER プロセスを再起動する。 \$ @SYS\$STARTUP:DECDTM\$STARTUP.COM 失敗 30 秒間待ってから手順 4b および 4c を繰り返す。

不可能な場合は、ノードをリブートしてトランザクション・ログを閉じる。リブートされたらノードにログインする。

5. LMCP ユーティリティの CONVERT LOG コマンドを使用して、トランザクション・ログのサイズを変更する。

CONVERT LOG/SIZE= サイズ SYSTEM\$ ノード.LM\$OLD
ディレクトリ指定 SYSTEM\$ ノード.LM\$JOURNAL

サイズ 新しいトランザクション・ログのサイズを示すブロック数。
ディレクトリ指定 トランザクション・ログを格納するディレクトリの完全指定。
ノード トランザクション・ログが置かれているノードの名前。

6. ステップ 4 で DECdtm サービスを使用しているソフトウェアを停止した場合は、そのソフトウェアを再起動する。

7. 旧トランザクション・ログを削除する。
DELETE ディレクトリ指定 SYSTEM\$ ノード.LM\$OLD;

ディレクトリ指定 旧トランザクション・ログを格納しているディレクトリの完全指定。
ノード トランザクション・ログが置かれているノードの名前。

例

この例は、ノード RED のトランザクション・ログのサイズを 6000 ブロックに変更していません。ノード RED は OpenVMS Cluster 内に存在し、そのトランザクション・ログは DISK\$LOG2:[LOGFILES] に置かれています。

ノード RED にログインします。RED のトランザクション・ログが置かれているディレクトリを探し、そのトランザクション・ログの名前を変更します。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> SHOW LOG SYSTEM$RED.LM$JOURNAL
Directory of DISK$LOG2:[LOGFILES]
```

```
SYSTEM$RED.LM$JOURNAL;1
```

```
Total of 1 file.
```

```
LMCP> EXIT
```

```
$ RENAME DISK$LOG2:[LOGFILES] SYSTEM$RED.LM$JOURNAL -
_$ DISK$LOG2:[LOGFILES] SYSTEM$RED.LM$OLD
```

DECdtm サービスを使用しているすべてのソフトウェアを終了します。次に、トランザクション・ログを閉じます。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> CLOSE LOG
Transaction log closed, TP_SERVER process stopped
LMCP> EXIT
```

TP_SERVER プロセスを再起動します。

```
$ @ SYS$STARTUP:DECDTM$STARTUP.COM
```

トランザクション・ログのサイズを変更します。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> CONVERT LOG/SIZE=6000 DISK$LOG2:[LOGFILES] SYSTEM$RED.LM$OLD -
_LMCP> DISK$LOG2:[LOGFILES] SYSTEM$RED.LM$JOURNAL
Log file DISK$LOG2:[LOGFILES] SYSTEM$RED.LM$JOURNAL;1 created.
Log file DISK$LOG2:[LOGFILES] SYSTEM$RED.LM$OLD converted.
LMCP> EXIT
```

DECdtm サービスを使用するソフトウェアを再起動します。
旧トランザクション・ログを削除します。

```
$ DELETE DISK$LOG2:[LOGFILES] SYSTEM$RED.LM$OLD;
```

14.8 トランザクション・ログの移動

次の場合には、トランザクション・ログを移動します。

- トランザクション・ログをより高速なディスク上に置きたい。
- ディスク上の作業負荷をさらに分散させたい。

作業方法



重要:

以下の手順で示すステップのすべてを実行してください。途中のステップを省略すると、データが壊れることがあります。

1. 14.2.2 項「トランザクション・ログの格納場所の決定」で示したガイドラインに従って、トランザクション・ログの移動先を決定する。ディスクにはトランザクション・ログを格納するのに十分な連続領域が存在する必要がある。
2. トランザクション・ログが置かれているノードにログインする。
3. OpenVMS Cluster 環境の場合、トランザクション・ログを移動するディスクが、クラスタ全体でマウントされていることを確認する。

4. トランザクション・ログの移動先のディレクトリを決定する。必要であれば、トランザクション・ログ専用のディレクトリを作成する。
5. LMCP の SHOW LOG コマンドを使用して、トランザクション・ログが置かれているディレクトリを探す。
SHOW LOG SYSTEM\$ ノード.LM\$JOURNAL
ノードは、トランザクション・ログが置かれているノードの名前。
6. トランザクション・ログをリネームする。
RENAME ディレクトリ指定 SYSTEM\$ ノード.LM\$JOURNAL
ディレクトリ指定 SYSTEM\$ ノード.LM\$OLD

ディレクトリ指定 トランザクション・ログを格納するディレクトリの完全指定。
ノード トランザクション・ログが置かれているノードの名前。
7. ノードを全くシャットダウンしないで DECdtm サービスを使用しているすべてのソフトウェアを停止できるか。
可能な場合は、次のようにトランザクション・ログを閉じる。

手順	作業
a.	DECdtm サービスを使用しているすべてのソフトウェアを終了する。
b.	LMCP の CLOSE LOG コマンドを使用して、トランザクション・ログを閉じる。 <pre>\$ RUN SYS\$SYSTEM:LMCP LMCP> CLOSE LOG</pre> CLOSE LOG コマンドはトランザクション・ログを閉じてから DECdtm TP_SERVER プロセスを終了する。DECdtm サービスを使用しているソフトウェアが 1 つでもあれば コマンドは失敗する。
c.	CLOSE LOG コマンドの実行に成功したか。 成功 TP_SERVER プロセスを再起動する。 <pre>\$ @SYS\$STARTUP:DECDTM\$STARTUP.COM</pre> 失敗 30 秒間待ってから手順 7b および 7c を繰り返す。

不可能な場合は、ノードをリブートしてトランザクション・ログを閉じる。リブートされたらノードにログインする。

8. 論理名 SYS\$JOURNAL に、ログの移動先のディレクトリが定義されていることを確認する。定義されていない場合は、SYS\$JOURNAL を再定義する。
DO DEFINE/SYSTEM/EXECUTIVE_MODE SYS\$JOURNAL ディレクトリ指定 [...]
ここで、**ディレクトリ指定** は 1 つまたは複数のトランザクション・ログを格納するディレクトリの完全ファイル指定である。トランザクション・ログの移動後にトランザクション・ログを格納するすべてのディレクトリを指定する。ディレクトリの順序は自由。
OpenVMS Cluster 環境では、SYSMAN を使用して SYS\$JOURNAL をクラスタ全体で再定義する必要がある。
9. ステップ 8 で SYS\$JOURNAL を再定義した場合は、それに合わせてコマンド・プロシージャ SYS\$MANAGER:SYLOGICALS.COM の中の SYS\$JOURNAL の定義を更新する。
ノード独自の SYLOGICALS.COM を作成している場合は、そのコマンド・プロシージャのすべてを更新する。
10. LMCP ユーティリティの CONVERT LOG コマンドを使用して、トランザクション・ログを移動する。

CONVERT LOG 古いディレクトリ指定 SYSTEM\$ ノード.LM\$OLD
新しいディレクトリ指定 SYSTEM\$ ノード.LM\$JOURNAL

古いディレクトリ指定 現在トランザクション・ログを格納しているディレクトリの完全指定。
ノード トランザクション・ログが置かれているノードの名前。
新しいディレクトリ指定 トランザクション・ログの移動先のディレクトリの完全指定。

11. ステップ7でDECdtm サービスを使用しているソフトウェアを停止した場合は、そのソフトウェアを再起動する。
12. 旧トランザクション・ログを削除する。
DELETE ディレクトリ指定 SYSTEM\$ ノード.LM\$OLD;

ディレクトリ指定 トランザクション・ログを格納するディレクトリの完全指定。
ノード トランザクション・ログが置かれているノードの名前。

例

この例は、BLUE のトランザクション・ログの移動方法を紹介しています。BLUE は OpenVMS Cluster 内に存在します。クラスタ・メンバおよびトランザクション・ログの格納場所を次のように想定しています。

ノード	ログを格納するディレクトリ
BLUE	DISK\$LOG1:[LOGFILES]
RED	DISK\$LOG2:[LOGFILES]

どちらのノードも独自の SYLOGICALS.COM を使用していないものとします。

BLUE のトランザクション・ログの移動先を決定します。この例では、DISK\$LOG3:[LOGFILES] に移動します。

ノード BLUE にログインします。次にディスクをクラスタ全体でマウントしてから、トランザクション・ログ用に新しいディレクトリを作成します。

```
$ MOUNT/CLUSTER/SYSTEM DUA3: LOG3  
$ CREATE/DIRECTORY DISK$LOG3:[LOGFILES]
```

BLUE のトランザクション・ログが置かれているディレクトリを探し、トランザクション・ログの名前を変更します。

```
$ RUN SYS$SYSTEM:LMCP  
LMCP> SHOW LOG SYSTEM$BLUE.LM$JOURNAL  
Directory of DISK$LOG1:[LOGFILES]
```

```
SYSTEM$BLUE.LM$JOURNAL;1
```

```
Total of 1 file.
```

```
LMCP> EXIT
```

```
$ RENAME DISK$LOG1:[LOGFILES] SYSTEM$BLUE.LM$JOURNAL -  
_ $ DISK$LOG1:[LOGFILES] SYSTEM$BLUE.LM$OLD
```

DECdtm サービスを使用しているすべてのソフトウェアを終了します。次にトランザクション・ログを閉じます。

```
$ RUN SYS$SYSTEM:LMCP  
LMCP> CLOSE LOG
```

```
Transaction log closed, TP_SERVER process stopped
LMCP> EXIT
```

TP_SERVER プロセスを再起動します。

```
$ @SYS$STARTUP:DECDTM$STARTUP.COM
```

SYS\$JOURNAL を再定義します。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> SET ENVIRONMENT/CLUSTER
SYSMAN> DO DEFINE/SYSTEM/EXECUTIVE_MODE SYS$JOURNAL -
SYSMAN> DISK$LOG2:[LOGFILES], DISK$LOG3:[LOGFILES]
SYSMAN> EXIT
```

コマンド・プロシージャ SYS\$MANAGER:SYLOGICALS.COM の SYS\$JOURNAL の定義内容を更新します。その後、トランザクション・ログを移動します。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> CONVERT LOG DISK$LOG1:[LOGFILES] SYSTEM$BLUE.LM$OLD -
LMCP> DISK$LOG3:[LOGFILES] SYSTEM$BLUE.LM$JOURNAL
Log file DISK$LOG3:[LOGFILES] SYSTEM$BLUE.LM$JOURNAL;1 created.
Log file DISK$LOG1:[LOGFILES] SYSTEM$BLUE.LM$OLD converted.
LMCP> EXIT
```

DECdtm サービスを使用するソフトウェアを再起動します。次に旧トランザクション・ログを削除します。

```
$ DELETE DISK$LOG1:[LOGFILES] SYSTEM$BLUE.LM$OLD;
```

14.9 ディスクのディスマウント

ディスクをディスマウントするには、ディスク上のトランザクション・ログをすべて閉じておく必要があります。

ここでは、トランザクション・ログを持っているディスクのディスマウント方法を説明します。

作業方法

1. LMCP の SHOW LOG コマンドを使用して、ディスマウントしたいディスク上に置かれているトランザクション・ログを探す。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> SHOW LOG
```

2. ノードを全くシャットダウンしないで、DECdtm サービスを使用しているすべてのソフトウェアを終了する。

ソフトウェアを終了できなければ、ステップ 3 で 1 つまたは複数のノードをリブートする必要はある。

3. ディスク上の各トランザクション・ログに対して次のステップを実行する。
 - a. トランザクション・ログが置かれているノードにログインする。
 - b. トランザクション・ログの名前を変更する。
RENAME ディレクトリ指定 SYSTEM\$ ノード.LM\$JOURNAL
ディレクトリ指定 SYSTEM\$ ノード.LM\$TEMP

ディレクトリ指定 トランザクション・ログを格納するディレクトリの完全指定。

ノード トランザクション・ログを格納するノードの名前。

- c. DECdtm サービスを使用しているすべてのソフトウェアをステップ 2 で終了したか。終了済みの場合は、次のようにトランザクション・ログを閉じる。

手順	作業
1)	LMCP の CLOSE LOG コマンドを使用してトランザクション・ログを閉じる。 <pre>\$ RUN SYS\$SYSTEM:LMCP LMCP> CLOSE LOG</pre> CLOSE LOG コマンドはトランザクション・ログを閉じてから DECdtm TP_SERVER プロセスを終了する。DECdtm サービスを使用しているソフトウェアが 1 つでもあれば コマンドは失敗する。
2)	CLOSE LOG コマンドの実行に成功したか。 成功 TP_SERVER プロセスを再起動する。 <pre>\$ @SYS\$STARTUP:DECDTM\$STARTUP.COM</pre> 失敗 30 秒間待ってから手順 3c を繰り返す。

未終了の場合は、ノードをリブートしてトランザクション・ログを閉じる。リブートされたらノードにログインする。

- d. ディスクをディスマウントする。ディスクのディスマウントに関しては、14.9 項「ディスクのディスマウント」を参照。
4. ディスクを再度マウントしたいときは、次のステップを実行する。
- a. ディスクをマウントする。ディスクのマウントに関しては、『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル (上巻)』を参照。
クラスタの場合は、クラスタ全体でディスクをマウントする。
- b. ディスク上の各トランザクション・ログの名前を変更する。
 RENAME ディレクトリ指定 SYSTEM\$ ノード.LM\$TEMP
 ディレクトリ指定 SYSTEM\$ ノード.LM\$JOURNAL
- ディレクトリ指定** トランザクション・ログを格納するディレクトリの完全指定。
ノード トランザクション・ログを格納するノードの名前。
- c. DECdtm サービスを使用するソフトウェアを終了した場合は、そのソフトウェアを再起動する。

例

次の例は、ディスク DISK\$LOG3 のディスマウントの方法を説明しています。ディスク上にあるトランザクション・ログを探す。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> SHOW LOG
```

```
Directory of DISK$LOG3: [LOGFILES]
```

```
SYSTEM$BLUE.LM$JOURNAL;1
```

DISK\$LOG3 上に存在するトランザクション・ログは、ノード BLUE のトランザクション・ログだけである。

DECdtm サービスを使用するすべてのソフトウェアを終了する。

ノード BLUE にログインする。次にトランザクション・ログの名前を変更する。

```
$ RENAME DISK$LOG3:[LOGFILES] SYSTEM$BLUE.LM$JOURNAL -
_$ DISK$LOG3:[LOGFILES] SYSTEM$BLUE.LM$TEMP
トランザクション・ログを閉じる。
```

```
$ RUN SYS$SYSTEM:LMCP
LMCP> CLOSE LOG
Transaction log closed, TP_SERVER process stopped
LMCP> EXIT
TP_SERVER プロセスを再起動する。
```

```
$ @SYS$STARTUP:DECDTM$STARTUP.COM
ディスクをディスマウントする。
```

```
$ DISMOUNT/CLUSTER DISK$LOG3:
ディスクを再度マウントしたいときは、クラスタ全体でマウントする。
```

```
MOUNT/CLUSTER/SYSTEM DUA3: LOG3
BLUE のトランザクション・ログの名前を変更する。
```

```
$ RENAME DISK$LOG3:[LOGFILES] SYSTEM$BLUE.LM$TEMP -
_$ DISK$LOG3:[LOGFILES] SYSTEM$BLUE.LM$JOURNAL
DECdtm サービスを使用するソフトウェアを再起動する。
```

14.10 ノードの追加

OpenVMS Cluster に追加したすべてのノードで、新しいトランザクション・ログを作成する必要があります。この節では、新規ノードでのトランザクション・ログの作成方法について説明します。

作業方法

この作業を実行するには、あらかじめ新規ノードをクラスタ内で構成しておく必要があります。クラスタ内での新規ノードの構成方法については、『OpenVMS Cluster システム』を参照してください。

1. 14.2 項「トランザクション・ログの計画」のガイドラインを使用して、新規ノードのトランザクション・ログのサイズと格納場所を決定する。ディスクにはログを格納するのに十分な連続領域が存在する必要がある。
2. トランザクション・ログを作成するディスクが、クラスタ全体でマウントされていることを確認する。
3. 新規トランザクション・ログを作成するディレクトリを決定する。必要であれば、トランザクション・ログ専用のディレクトリを作成することもできる。
4. SYS\$JOURNAL に新しいノードのトランザクション・ログを格納するディレクトリが定義されていることを確認する。定義されていない場合は、SYSMAN を使用して SYS\$JOURNAL をクラスタ全体で再定義する。

```
DO DEFINE/SYSTEM/EXECUTIVE_MODE SYS$JOURNAL ディレクトリ指定 [...]
```

ディレクトリ指定は、トランザクション・ログを格納するディレクトリの完全指定である。トランザクション・ログを含むすべてのディレクトリ(新規ノードのトランザクション・ログを作成するディレクトリも含める)を指定する。ディレクトリの指定順序は自由である。

- ステップ 4 で SYS\$JOURNAL を再定義した場合は、SYS\$MANAGER:SYLOGICALS.COM コマンド・プロシージャを変更して SYS\$JOURNAL 定義を更新する。
ノード固有の SYLOGICALS.COM を作成した場合は、すべての SYLOGICALS.COM を変更する。
- LMCP の CREATE LOG コマンドを使用してトランザクション・ログを作成する。
CREATE LOG [/SIZE= サイズ] ディレクトリ指定 SYSTEM\$ ノード.LM\$JOURNAL

サイズ	トランザクション・ログのサイズを示すブロック数。省略時の設定では 4000 ブロック。
ディレクトリ指定	トランザクション・ログを作成するディレクトリの完全指定。
ノード	新しいノードの名前。

例

この例は、SCSNODE 名が WHITE である新規ノードで、トランザクション・ログを作成する方法を説明したものです。

この例では、クラスタ・メンバおよびそのトランザクション・ログの格納場所は次のとおりです。

ノード	ログを格納するディレクトリ
BLUE	DISK\$LOG3:[LOGFILES]
RED	DISK\$LOG2:[LOGFILES]

どちらのノードもノード固有の SYLOGICALS.COM は持っていません。
WHITE のトランザクション・ログのサイズと格納場所を決定します。

ノード	ログのサイズ (ブロック数)	ディスク
WHITE	5000	DUA4

DUA4 をクラスタ全体でマウントします。

```
MOUNT/CLUSTER/SYSTEM DUA4: LOG4
```

トランザクション・ログ用のディレクトリを作成します。

```
$ CREATE/DIRECTORY DISK$LOG4: [LOGFILES]
```

SYS\$JOURNAL を再定義します。

```
$ RUN SYS$SYSTEM:SYSMAN
```

```
SYSMAN> SET ENVIRONMENT/CLUSTER
```

```
SYSMAN> DO DEFINE/SYSTEM/EXECUTIVE MODE SYS$JOURNAL -
```

```
_SYSMAN> DISK$LOG2: [LOGFILES], DISK$LOG3 [LOGFILES], DISK$LOG4: [LOGFILES]
```

```
SYSMAN> EXIT
```

SYS\$MANAGER:SYLOGICALS コマンド・プロシージャを編集して、SYS\$JOURNAL 定義を更新します。次にトランザクション・ログを作成します。

```
$ RUN SYS$SYSTEM:LMCP
```

```
LMCP> CREATE LOG/SIZE=5000 DISK$LOG4: [LOGFILES] SYSTEM$WHITE.LM$JOURNAL
```

```
LMCP> EXIT
```

14.11 ノードの削除

この節では、DECdtm サービスを使用している場合にノードを削除する方法について説明します。

作業方法

スタンドアロン・マシンを使用している場合は、ステップ 1 ～ 8 までだけを実行します。



重要:

以下の手順で示すステップのすべてを実行してください。途中のステップを省略すると、データが壊れることがあります。

1. 削除したいノードにログインする。
2. DECdtm サービスを使用しているすべてのソフトウェアを終了する。
3. LMCP の DUMP/ACTIVE コマンドを使用して、ノードのトランザクション・ログに実行中のトランザクションが含まれているかどうかを確認する。
DUMP/ACTIVE SYSTEM\$ ノード.LM\$JOURNAL
ノードは削除したいノードの名前である。
このコマンドはすべての実行中のトランザクションの詳細を表示する。最後の行には実行中のトランザクションの総数が表示される。
4. トランザクション・ログに実行中のトランザクションが含まれている場合は、以下のステップを実行する。
 - a. DECdtm サービスを使用しているすべてのソフトウェアで復旧プロシージャを実行する。
 - b. LMCP の DUMP/ACTIVE コマンドを使用して、ノードのトランザクション・ログに実行中のトランザクションがまだ含まれているかどうかを確認する。
 - c. トランザクション・ログに実行中のトランザクションがまだ含まれている場合は、弊社のサポート担当者に連絡する。
5. SYS\$JOURNAL を再定義し、削除したいノードのトランザクション・ログを格納しているディレクトリを削除する。ただし、ディレクトリに他のトランザクション・ログが含まれている場合を除く。
DEFINE/SYSTEM/EXECUTIVE_MODE SYS\$JOURNAL ディレクトリ指定 [,...]
ここで、**ディレクトリ指定** は 1 つまたは複数のトランザクション・ログを含むディレクトリの完全指定。削除したいノードのトランザクション・ログを含むディレクトリだけでなく、トランザクション・ログを格納するすべてのディレクトリを指定する。ディレクトリの指定順序は自由である。
クラスタの場合は、SYSMAN を使用して SYS\$JOURNAL をクラスタ全体で再定義する。
6. ステップ 5 で SYS\$JOURNAL を再定義した場合は、コマンド・プロシージャ SYS\$MANAGER:SYLOGICALS.COM の SYS\$JOURNAL の定義を更新する。
独自の SYLOGICALS.COM を使用しているノードがある場合は、そのようなノードのすべてでコマンド・プロシージャを更新する。
7. トランザクション・ログを保管する。
8. ノードをシャットダウンする。
9. DECdtm サービスを使用するソフトウェアを再起動する。
10. クラスタを再構成してノードを削除する。
クラスタの再構成に関しては、『OpenVMS Cluster システム』を参照。

例

この例は、ノード BLUE の削除方法を示しています。クラスタ・メンバおよびトランザクション・ログの格納場所を次のように想定しています。

ノード	ログを格納するディレクトリ
BLUE	DISK\$LOG3:[LOGFILES]
RED	DISK\$LOG2:[LOGFILES]
WHITE	DISK\$LOG4:[LOGFILES]

どのノードもノード固有の SYLOGICALS.COM コマンド・プロシージャは持っていません。

ノード BLUE にログインします。

DECdtm サービスを使用しているすべてのソフトウェアを終了します。BLUE のトランザクション・ログに実行中のトランザクションが含まれているかどうかを確認します。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> DUMP/ACTIVE SYSTEM$BLUE.LM$JOURNAL
Dump of log file DISK$LOG3:[LOGFILES]SYSTEM$BLUE.LM$JOURNAL

Total of 0 transactions active, 0 prepared and 0 committed.
LMCP> EXIT
```

SYS\$JOURNAL を再定義します。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> SET ENVIRONMENT/CLUSTER
SYSMAN> DO DEFINE/SYSTEM/EXECUTIVE_MODE SYS$JOURNAL -
SYSMAN> DISK$LOG2:[LOGFILES], DISK$LOG4:[LOGFILES]
SYSMAN> EXIT
```

SYS\$MANAGER:SYLOGICALS.COM コマンド・プロシージャを編集して SYS\$JOURNAL 定義を更新します。

BLUE のトランザクション・ログを保管し、ノード BLUE をシャットダウンします。

```
$ @SYS$SYSTEM:SHUTDOWN.COM
```

Should an automatic system reboot be performed [NO]? **NO**

DECdtm サービスを使用するソフトウェアを再起動します。その後、クラスタを再構成します。

```
$ @SYS$STARTUP:CLUSTER_CONFIG.COM
```

Cluster Configuration Procedure

1. ADD a node to a cluster.
2. REMOVE a node from the cluster.
3. CHANGE a cluster member's characteristics.
4. CREATE a duplicate system disk for BLUE.

Enter choice [1]: **2**

Updating network database...

The configuration procedure has completed successfully.

14.12 DECdtm サービスの停止

省略時の設定では、システムをブートすると DECdtm サービスが自動的に開始され、トランザクション・ログが 1 つ 見つかるまで DECdtm プロセス TP_SERVER がチェックします。

DECdtm サービスを利用するソフトウェアを現在使用しないか、使用する予定がない場合は、DECdtm サービスを停止します。これでメモリと CPU 時間を節約することができます。

OpenVMS Cluster 内ではすべてのノードで DECdtm サービスを停止します。

作業方法

1. 各ノードで以下を実行する。
 - a. ノードにログインする。
 - b. LMCP の CLOSE LOG コマンドを使用して TP_SERVER プロセスを停止する。

```
$ RUN SYS$SYSTEM:LMCP
LMCP> CLOSE LOG
```

CLOSE LOG コマンドは、どのソフトウェアも DECdtm サービスを使用していなければ、TP_SERVER プロセスを終了する。

CLOSE LOG コマンドが実行されなかった場合は作業を中断する。クラスタ・システム内の別のノードで TP_SERVER プロセスの終了を完了している場合は、SYS\$STARTUP:DECDTM\$STARTUP.COM コマンド・プロシージャを使用してそのプロセスを再起動する。

2. 次の行を SYS\$MANAGER:SYLOGICALS.COM コマンド・プロシージャに追加する。

```
$ !
$ DEFINE/SYSTEM/EXECUTIVE_MODE SYS$DECDTM_INHIBIT yes
$ !
```

固有の SYLOGICALS.COM を使用しているノードがある場合は、そのすべてのプロシージャを変更する。

これにより TP_SERVER プロセスは、システムの次のブート時から作成されなくなる。

14.13 DECdtm サービスの開始

ここで示す作業が必要になるのは、DECdtm サービスをいったん停止した後、DECdtm サービスを利用するソフトウェアを実行するようになった場合だけです。

作業方法

1. 論理名 SYS\$DECDTM_INHIBIT の指定を解除する。

```
$ DEASSIGN/SYSTEM/EXECUTIVE_MODE SYS$DECDTM_INHIBIT
```

OpenVMS Cluster 環境では、SYSMAN を使用して SYS\$DECDTM_INHIBIT をクラスタ全体で指定解除する。

2. DECdtm サービス・プロセス TP_SERVER を起動する。

```
$ @SYS$STARTUP:DECDTM$STARTUP.COM
```

OpenVMS Cluster 環境では、SYSMAN を使用して TP_SERVER プロセスをクラスタ全体で起動する。

3. コマンド・プロシージャ SYS\$MANAGER:SYLOGICALS.COM から、SYS\$DECDTM_INHIBIT の定義を削除する。これにより、次のブート時から DECdtm サービスが自動的に起動されるようになる。

例

この例は、DECdtm サービスをクラスタ環境で開始する方法を説明しています。

SYS\$DECDTM_INHIBIT の指定を解除してから TP_SERVER プロセスを起動します。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> SET ENVIRONMENT/CLUSTER
SYSMAN> DO DEASSIGN/SYSTEM/EXECUTIVE_MODE SYS$DECDTM_INHIBIT
SYSMAN> DO @SYS$STARTUP.DECDTM$STARTUP.COM
SYSMAN> EXIT
```

SYS\$MANAGER:SYLOGICALS.COM コマンド・プロシージャを編集して SYS\$DECDTM_INHIBIT 定義を削除します。

制限事項

DECdtm は、作業を実行するために必要なオペレーティング環境がない場合、クラッシュの原因となる重大なシステム障害 (BUGCHECK) を引き起こします。一般的な例は、仮想メモリの割り当ての障害です。このようなエラーは、DECdtm の重大な例外として扱われるため、BUGCHECK で失敗し、結局はシステムがクラッシュします。

DECdtm は、主にそのデータ構造に依存して、トランザクションの完全性と一貫性を維持しています。そのため、実行中にエラーや致命的な状況が発生すると、重大な矛盾として扱われます。この結果、DECdtm は致命的な例外が発生させ、システムをクラッシュさせます。

オペレーティング環境では、仮想メモリや非ページング・プールなどのシステム・リソースに十分なバッファを確保することをお勧めします。DECdtm を使用していない環境でクラッシュが発生した場合は、DECdtm を起動しないようにしてください。(DECdtm を起動しないようにする方法については、14.12 項「DECdtm サービスの停止」を参照してください。)

164 プラットフォームでの「動的メモリ不足」に対処するための DECdtm 論理名の定義

KPB (Kernel Process Block) のメモリ割り当て中に発生する動的メモリ不足に対処するために、DECdtm 論理名が用意されています。この設計ロジックは、リトライ・ロジックと呼ばれています。このリトライ・ロジックでは、システム・テーブルの論理名 SYS\$DECDTM_KPALLOC_RETRYCNT を定義して、システム・サービス EXE\$KP_ALLOCATE_KPB の呼び出しで戻り値が SS\$_INSFMEM の場合にこのサービスを繰り返す回数 (リトライ・カウント) を設定できます。

システム論理名 SYS\$DECDTM_KPALLOC_RETRYCNT には、オプション文字列リテラル "MIN", "DEF", または "MAX" を設定します。この文字列リテラルは、リトライ・カウント MIN=25, DEF=50, および MAX=100 を示します。

論理名 SYS\$DECDTM_KPALLOC_RETRYCNT がシステム・テーブルに見つからない場合、リトライ・ロジックでは省略時のカウント DEF=50 が使用されます。

例

```
$ DEFINE/SYS/EXEC SYS$DECDTM_KPALLOC_RETRYCNT MAX
```

14.14 XA Gateway の使用 (Alpha および I64)

DECdtm/XA は、XA を使用するトランザクションを調整し、管理するためのサポートを提供します。XA Gateway を使用することにより、DECdtm/XA は、他のトランザクション・マネージャ (TM) によって管理されるトランザクション内の他のリソース・マネージャ (RM) を参加させることができます。このセクションでは、DECdtm XA Gateway サポートを設定し、使用方法について説明します。



注意:

この章では、**XA Specification** は Distributed Transaction Processing: The XA Specification を意味します。

DECdtm/XA を使用し、DECdtm/XA サービスのスタートアップおよびシャットダウンが確実に正しく実行されるようにするには、以下のファイルを起動する必要があります。

- SYS\$STARTUP:DDTM\$XA_STARTUP.COM
- SYS\$STARTUP:DDTM\$XA_SHUTDOWN.COM

コマンド @SYS\$STARTUP:DDTM\$XA_STARTUP.COM を、スタートアップ・データベースまたはコマンド・ファイル SYS\$MANAGER:SYSTARTUP_VMS.COM に追加します。

コマンド @SYS\$STARTUP:DDTM\$XA_SHUTDOWN.COM をコマンド・ファイル SYS\$MANAGER:SYSHUTDWN.COM に追加します。

以下の手順を実行して、DECdtm XA サービスが正しく実行されていることを確認します。

1. XGCP ユーティリティを使用して、ローカルの OpenVMS ノードと同じ名前のゲートウェイ・ログ・ファイルを作成する。詳細は、14.14.1 項「ゲートウェイの設定」および『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照。
2. SYS\$TEST:DECDTM_XG_IVP.EXE を実行する。
3. XGCP ユーティリティを使用して、ゲートウェイ・サーバを終了し、再起動する。ゲートウェイにローカルの OpenVMS ノードと異なる名前をつけて設定する場合には、この手順が必須である。XGCP ユーティリティについての詳細は、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル (下巻)』を参照。

14.14.1 ゲートウェイの設定



注記: HP DECdtm/XA Version 2.1 Gateway には現在、クラスタ単位のトランザクション回復サポートがあります。クラスタ単位の DECdtm Gateway Domain Log を使用するアプリケーションのトランザクションは、どのような単一ノード障害からも回復できるようになりました。障害が発生したノードの代わりに、残っているクラスタ・ノードで動作しているゲートウェイ・サーバが、トランザクション回復処理を起動できます。

XA Gateway は、XA 対応のリソース・マネージャとして、各トランザクション処理 (TP) プロセスに設定されます。XA Gateway は XA トランザクション・マネージャ (TM) からの XA 呼び出しを処理し、これらの呼び出しを DECdtm システム・サービスの呼び出しにマップします。これにより、DECdtm は、1 つの TP プロセスで使用されるどの DECdtm 対応のリソース・マネージャ (RM) に対しても、正しいイベントを送信できるようになります。

XA Gateway の操作は、RM からは透過的です。このため、DECdtm の RM は、何も変更することなく XA Gateway で使用することができます。

XA Gateway では、ログ・ファイルを使用して、XA のトランザクションと DECdtm のトランザクションとのマッピングを記録します。このログ・ファイルは、ゲートウェイ・サーバ・プロセス DDTM\$XG_SERVER によって管理されます。

ゲートウェイ・ログ・ファイルは、XGCP ユーティリティを使用して作成します (『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照)。ゲートウェイ・ログ・ファイルのサイズは、同時にアクティブになっているトランザクションの数によって決まりません。サイズは、XA TM が使用するトランザクション ID (TID) のサイズによって異なりますが、アクティブな各トランザクションごとに、最大で 600 バイトが必要です。このゲートウェイ・ログ・ファイルは、必要に応じて自動的に拡張されます。

ゲートウェイ・ログ・ファイルは、論理名 SYS\$JOURNAL によって指定されたディレクトリに常駐し、フォームの名前は SYSTEM\$name.DDTM\$XG_JOURNAL になります。性能の最適化のために、各ゲートウェイ・ログ・ファイルおよび DECdtm ログ・ファイルを、個別の物理デバ

イスに移動し、物理デバイスのセットに対する検索リストとして SYS\$JOURNAL を定義します。

XA Gateway では、各 OpenVMS Cluster ノードごとに、XA トランザクション・マネージャと XA Gateway ログ・ファイルとの間に関連付けが必要です。この関連付けは、ゲートウェイ名を以下のように指定することによって管理されます。

- XGCP ユーティリティを使用して、ゲートウェイ名と同じ名前のゲートウェイ・ログ・ファイルを作成する (『OpenVMS システム管理 ユーティリティ・リファレンス・マニュアル』を参照)。
- ゲートウェイ名は、XA TM の制御下で実行されるアプリケーション内で Gateway RM が設定されるときに、xa_open 情報文字列内で指定される (XA RM の設定については、『OpenVMS Programming Concepts Manual, Volume II』を参照)。
- XA TM によって実行される最初の XA アプリケーションにより、ゲートウェイ名が OpenVMS Cluster のローカル・ノードに結び付けられる。ゲートウェイ・サーバが終了するまで、ゲートウェイ名は、そのノードに結び付けられたままになる。

同じローカル・ノードで実行されるすべての XA アプリケーションは、同じゲートウェイ名を使用して設定する必要があります。同じ名前を使用している複数の XA アプリケーションを、他の OpenVMS Cluster ノードで実行することはできません。したがって、通常は OpenVMS Cluster の各ノードごとに、1つのゲートウェイ名を定義し、1つのゲートウェイ・ログ・ファイルを作成します。

ゲートウェイ名の関連付けを変更して、ゲートウェイ名を別の OpenVMS Cluster ノードに結び付けることができます。ただし、そのノードがゲートウェイ・ログ・ファイルにアクセスできることが必要条件です。ゲートウェイ名の関連付けを変更するには、以下の手順を実行します。

1. 元のノードで実行中のすべての XA アプリケーションを終了する。
2. XGCP ユーティリティを使用して、元のノードで実行中のゲートウェイ・サーバを終了する。
3. 新しいノードで実行中のすべての XA アプリケーションを終了する。
4. 新しいノードで実行中のゲートウェイ・サーバを終了し、ゲートウェイを再起動する。
5. 新しいノードで、元の XA アプリケーションを実行する。



注意:

ゲートウェイ・ログ・ファイルを損失から保護するため、常駐しているデバイスのシャドウイングを実行するなどの対策を講じる必要があります。新しいログ・ファイルを生成したり、古いログ・ファイルを使用したりすると、元の記録ではすでにコミットしたことになるトランザクションが、間違っただけでロール・バックされる可能性があります。これは、データベースの間で相互に一貫性が失われたり、他のシステムまたはユーザに送信されるレポートとの一貫性が失われたりする原因になります。

ゲートウェイ・ログ・ファイルはサイズが大きいので、できるだけ削除しないようにしてください。不要なゲートウェイ・ログ・ファイルをどうしても削除する場合には、まず DECdtm XGCP ユーティリティを使用して、このゲートウェイがもうどの用意されたトランザクションの参加者にもなっていないことを確認します。このゲートウェイの参加者名は DDTM\$XG/name です。

ゲートウェイ・サーバは、以下のシステム論理名を使用します。

- SYS\$DECDTM_XG_REQS

サーバによって処理される同時要求の数で、範囲は 100 ~ 100,000 です。この数により、DDTM\$XG がサーバとやりとりするために使用するグローバル・セッションのサイズと、サーバで必要とされる制限値が決まります。このパラメータは、論理名 SYS\$DECDTM_XG_REQS を定義することによって指定されます。このパラメータへの変更は、サーバとすべてのクライアント・プロセスが終了されるまで、有効になりません。

処理中にこのパラメータの値が制限値を超えた場合には、クライアント要求は、並行して処理されるのではなく、ブロックされます。

- `SYS$DECDTM_XA_TRANS`

XA の同時トランザクションの予想される数の範囲は、1000 ~ 1,000,000 です。これにより、サーバ内部で使用されるインデックス・テーブルのサイズが決まります。このパラメータは、論理名 `SYS$DECDTM_XA_TRANS` を定義することによって指定されます。このパラメータへの変更は、サーバが終了されるまで有効になりません。

処理中にこのパラメータの値が制限値を超えた場合には、サーバによる CPU の使用量が増加します。しかしながら、その影響は、このパラメータの値が制限値の 10 倍以上にならないと目立ちません。

付録A Files-11 ディスク構造

この付録では、ディスクの概念と用語について説明します。予約ファイルについても解説し、ANALYZE/DISK_STRUCTURE ユーティリティで使用するファイルを紹介して、Files-11 オン・ディスク構造 (ODS) レベル 1, 2, 5 の違いについて述べます。以降の説明において、Files-11 ディスク構造とは Files-11 オン・ディスク構造のことを指します。

A.1 ディスクの概念

この節では、ディスクの物理的構造と論理的構造に関する用語について説明します。

A.1.1 ディスクの論理的構造

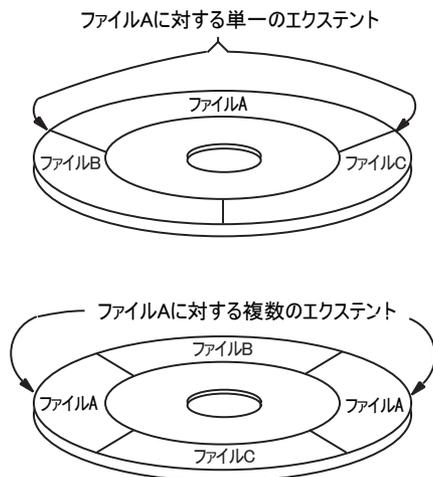
ディスクに記録されるアドレス可能な最小情報単位を、**ブロック** と呼びます。Files-11 ディスク構造のブロック 1 つは、512 バイト (1 バイト = 8 ビット) で構成されます。Files-11 ディスク・ボリュームとメモリとの間では、ブロック単位で転送を行うことができます。ただし、Files-11 ODS において、1 つのディスクは複数のブロックで構成される 1 つの配列であるとみなし、ブロック単位で扱うことは通常ありません。

ブロックは、論理的に**クラスタ** にまとめられます。ディスク空間は、このクラスタ単位で割り当てられます。**ボリューム** と呼ばれるディスクを使用可能状態にするとき、つまり初期化したときに、クラスタに入れるブロックの数を指定します。クラスタ・サイズは、媒体の種類によって異なりますが、サイズが小さいほど実用的です。一般的に、ブロック数が少ないディスクには小さいクラスタ・サイズを指定し、ブロック数が多いディスクには大きいクラスタ・サイズを指定して、ディスク空間を割り当てる際のオーバーヘッドを抑えるようにします。

1 つのファイルに割り当てられた連続するクラスタを、**エクステント** と呼びます。エクステントは、ファイル全体であることもファイルの一部であることもあります。連続するエリアがディスクに十分に存在する場合、1 つのエクステントとしてファイル全体を割り当てます。一方、ファイル全体を入れるのに十分な量の連続エリアが存在しない場合もあります。また、ファイルを作成した時点では、必要な空間すべてを予約しない場合もあると思われます。いったん作成した後でファイルを拡張する場合、隣接するクラスタは、おそらく別のファイルにすでに割り当てられています。したがって、連続しないクラスタが拡張に使用されます。

ファイルを複数の部分に分割した場合、各部分が 1 つのエクステントとなります。この結果、**図 A-1 「ファイルのエクステント」** に示すとおり、1 つのファイルを構成するエクステントがディスク上の別々のエリアに位置することもあり得ます。ファイルの拡張は、自動的に行われ

図 A-1 ファイルのエクステント



ZK0738GE

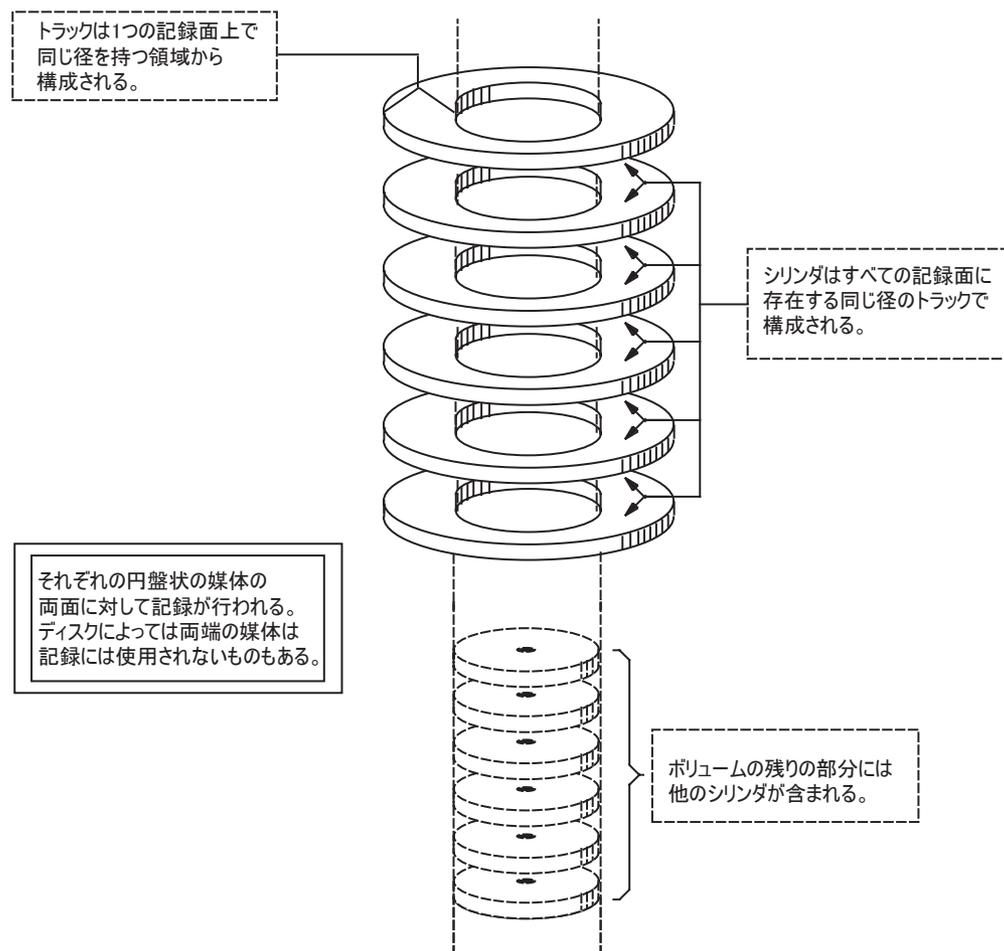
A.1.2 ディスクの物理的構造

Files-11 構造が認識する最小単位を、**セクタ**と呼びます。ほとんどの Files-11 ディスクの場合、セクタはブロックと同じ意味であり、512 バイトです。ディスクに関するその他の基本用語として、**トラック**と**シリンダ**があります。トラックとは、ディスクの記録面上で同じ径を持つセクタの集まりです (Files-11 構造ではブロックの集まり)。ディスク・デバイス上の特定の読み書きヘッド位置にアクセスすることができます。シリンダは、ディスクの全記録面に存在する、同じ径を持つ全トラックで構成されます。

あるシリンダのブロックにアクセスする場合、ディスクの読み書きヘッドを動かす必要はないので、関連性のあるデータ・ブロックを1つのシリンダにまとめると便利です。このため、大容量ディスクの場合には、シリンダ・サイズ単位で分割できるクラスタ・サイズを指定するようにしてください。

図 A-2 「トラックとシリンダ」は、トラックとシリンダを示しています。

図 A-2 トラックとシリンダ



ZK0740GE

A.2 Files-11 構造

Files-11 構造では、ボリュームまたはボリューム・セットを初期化したときに、削除不可能な予約ファイルの集合が作成されます。Files-11 ディスクの構造は、これらのファイルによって制御されます。Files-11 構造は、ディスク・パックなどの物理媒体であるボリュームに存在します。Files-11 ボリュームは、一定の順序で並ぶブロックの集合です (1 ブロック = 512 バイト)。ブロックには、0 から $n-1$ の番号が連続して付けられます。 $n-1$ は、ブロック単位によるディスク・サイズです。

A.2.1 ファイル識別子 (FID)

Files-11 ディスク上の各ファイルは、システムが割り当てて一意のファイル識別子 (FID) によって識別されます。また、ユーザが英数名を設定することもできます。Files-11 ディレクトリの主たる機能は、ユーザが設定した英数名を、対応する FID と対応づけることにあります。この結果、ファイルをファイル名で検索することができます。

ファイルの FID は、3 つの番号で構成されます。最初の番号は、**ファイル番号 (NUM)** です。ファイル・システムは、索引ファイル (予約ファイル INDEXF.SYS) へのオフセットとしてこの番号を使用します。索引ファイルには、ボリューム上の全ファイルに関する情報が入っています。

FID の 2 番目の部分は、**ファイル・シーケンス番号 (SEQ)** です。この番号は、ファイル番号を使用した回数を示します。ファイルを作成するとファイル番号が割り当てられ、ファイルを削除するとファイル番号の割り当てが解除されます。このため、ファイル番号だけではファイルを一意で識別することができません。ファイル番号を使用するたびにシーケンス番号を 1 ずつ増やすことにより、INDEXF.SYS に記録されている全ファイルの識別子を一意に保つことができます。

FID の 3 番目の番号は、**相対ボリューム番号 (RVN)** です。この番号は、ファイルが存在するボリューム (ボリューム・セット内の) を指します (ODS-2 専用)。ボリューム・セットに属するボリュームが 1 つだけである場合、このボリュームに存在するファイルの RVN は、すべて 0 です。

A.2.2 ODS ディレクトリ階層

Files-11 ODS-2 構造は、マルチレベルのディレクトリ階層です。ディレクトリ構造の最上位レベルは、マスタ・ファイル・ディレクトリ (MFD) です。ボリュームの MFD は、必ず [000000] と名付けられます。MFD には、MFD をはじめとする最上位レベルのディレクトリすべてと、予約ファイルが入っています。

ディレクトリとは、他のファイルが入っている 1 つのファイルです。ディレクトリに入っているファイルも、他のファイルを入れるディレクトリとなることができます。ディレクトリをネストすることにより、マスタ・ファイル・ディレクトリを含めて 9 レベルまでのディレクトリ階層を作ることができます。

ボリューム・セットの場合、ボリューム・セットに存在する全ユーザ・ディレクトリの MFD が、相対ボリューム 1 に存在します。この MFD のエントリは、ボリューム・セット内の任意のボリュームに存在するディレクトリを指し、これらのディレクトリは、ボリューム・セット内の任意のボリュームに存在するファイルとサブディレクトリを指します。残りのボリュームの MFD には、そのボリュームの予約ファイルの名前だけが入っています。

VAX システムの場合、Files-11 ODS-1 構造は、2 レベルのディレクトリ階層をサポートします。各 UIC (利用者識別コード) は、ユーザ・ファイル・ディレクトリ (UFD) と対応づけられます。各 UFD は、ボリュームのマスタ・ファイル・ディレクトリ (MFD) に入ります。

A.3 予約ファイル

この項では、Files-11 で使用する予約ファイルについて説明します。予約ファイルの FID は、すべて定数です。

また、ANALYZE/DISK_STRUCTURE で使用するファイルについても説明します。ANALYZE/DISK_STRUCTURE は、これらのファイルがあるべき姿をメモリにコピーし、現在のバージョンと比較します。/REPAIR 修飾子を指定すると、この比較処理で矛盾点が見つかった場合、これらの矛盾点をレポートして、修正します。

表 A-1 「予約ファイル」は、Files-11 レベル 1, 2, 5 で使用する予約ファイルと、ANALYZE/DISK_STRUCTURE で使用するファイルの一覧です。

表 A-1 予約ファイル

予約ファイル	ファイル名	構造レベル 1 ¹	構造レベル 2,5	ANALYZE/DISK_STRUCTURE
索引ファイル	INDEXF.SYS;1	X	X	X
ストレージ・ビット マップ・ファイル	BITMAP.SYS;1	X	X	X
不良ブロック・ファイ ル	BADBLK.SYS;1	X	X	
マスタ・ファイル・ ディレクトリ	000000.DIR;1	X	X	X
コア・イメージ・ファ イル	CORIMG.SYS;1	X	X	
ボリューム・セット・ リスト・ファイル	VOLSET.SYS;1		X	X
継続ファイル	CONTIN.SYS;1		X	
バックアップ・ログ・ ファイル	BACKUP.SYS;1		X	
保留不良ブロック	BADLOG.SYS;1		X	
クォータ・ファイル	QUOTA.SYS			X
ボリューム機密保護ブ ロファイル	SECURITY.SYS		X	

¹ VAX のみ

A.3.1 索引ファイル INDEXF.SYS

あらゆる Files-11 ボリュームに対し、ボリューム初期化時に索引ファイルが作成されます。Files-11 ディスクとしてディスクを使用するには、INITIALIZE コマンドで初期化する必要があります。

INDEXF.SYS は、いくつかのセクションで構成される、大型の拡張可能なファイルです。これらのセクションには、オペレーティング・システムのほか、Files-11 ボリュームの識別、ボリュームへの初期アクセス、ボリューム上の全ファイルの確認に必要な情報が入っています (INDEXF.SYS 自体も含む)。

表 A-2 「Files-11 索引ファイルの内容」は、INDEXF.SYS に入っている情報を示しています。ブート・ブロック、ホーム・ブロック、ファイル・ヘッダについては、後の項で詳しく説明します。

表 A-2 Files-11 索引ファイルの内容

用語	定義
ブート・ブロック	索引ファイルの仮想ブロック 1。ブート (またはブートストラップ) ・ブロックは、たいていはボリュームの論理ブロック 0 にマップされる。ボリュームがシステム・ボリュームの場合は、ブート・ブロックには、オペレーティング・システムをメモリにロードするプログラムが入っている。ボリュームがシステム・ボリュームではない場合は、ブート・ブロックには、ボリュームがシステム・デバイスではなく、ユーザのファイルだけが入っていることを示すメッセージを表示するプログラムが入っている。
ホーム・ブロック	ボリューム固有の ID が入っている。具体的には、ボリュームの名前と保護、ボリュームに格納できる最大ファイル数、ボリューム所有権情報である。ホーム・ブロックは、索引ファイルの仮想ブロック番号 2 である。
バックアップ・ホーム・ブロッ ク	ホーム・ブロックのコピー。1 次ホーム・ブロックが破壊されても、ボリュームを使用できるようにする。

表 A-2 Files-11 索引ファイルの内容 (続き)

用語	定義
バックアップ索引ファイル・ヘッダ	索引ファイル・ヘッダが壊れたときに、ボリューム上のデータを回復できるようにする。v * 3 + 1 から v * 4 の仮想ブロックを占有する。v は、ボリューム・クラスタの要素の数である。
索引ファイル・ビットマップ	ファイル・ヘッダの割り当てを制御し、結果的にボリューム上のファイル数を制御する。ファイル・ヘッダを示すビットが入っており、このビットが 0 である場合はファイルを作成でき、1 である場合はファイルがすでに使用されている。
ファイル・ヘッダ	索引ファイルの大部分を占める。索引ファイルへのアクセスに必要な全情報が入っている。1 つのファイル・ヘッダには、ボリュームに存在する 1 つのファイルに関する情報が入っている。情報の内容は、所有者 UIC、保護コード、作成日時、アクセス制御リスト (ACL) などである。また、ファイルを構成するエクステントのリストも入っており、ボリューム上の論理位置が示される。ファイル・ヘッダは、拡張ヘッダともなる。
索引ファイル代替ヘッダ	索引ファイルの 1 次ヘッダが破損したときに、データを回復できるようにする。

A.3.1.1 ブート・ブロック

システム・ディスクのブロック 0 は、**ブート・ブロック**です。ブート・ブロックには、システムのブート時に使用する**一次ブートストラップ・イメージ**の記憶位置とサイズが入っています。一部のプロセッサをブートする場合、ブート・ブロックを読み込んでブート・イメージの記憶位置を取り出す必要があります。詳細は、『OpenVMS システム管理者マニュアル (上巻)』のプロセス制御の章を参照してください。

A.3.1.2 ホーム・ブロック

ホーム・ブロックは、通常、ブート・ブロックの次のブロックです。ホーム・ブロックは、ディスクが Files-11 ボリュームであることを示します。ホーム・ブロックを読み込めない場合、つまり物理的に使用不能である場合は、別のブロックがホーム・ブロックとして選択されます。ボリューム自体の情報やボリューム上のファイルの省略時の値など、次のような項目がホーム・ブロックに入っています。

- ボリューム名
- 索引ファイルの他の部分を取り出すための情報
- 同時にボリュームに存在することができる最大ファイル数
- ボリュームの所有者の UIC (利用者識別コード)
- ボリューム全体を読み書きできるユーザを指定するボリューム保護情報

Files-11 ボリュームでは、ホーム・ブロックのコピーがいくつか用意されるので、ホーム・ブロックの情報を誤って消してしまうことを防止でき、ファイルへのアクセスも常に確保できます。

A.3.1.3 ファイル・ヘッダ

索引ファイルの大部分は、**ファイル・ヘッダ**です。1 つのファイル・ヘッダには、ボリュームに存在するファイルの内、1 つのファイルに関する情報が入っています。具体的には、所有者 UIC、保護コード、作成日時、ACL (アクセス制御リスト) などです。さらに、ファイルを構成するエクステントのリストが入っており、ボリューム上におけるファイルの論理位置が分かります。多数のエクステントで構成されるファイルの場合、複数のファイル・ヘッダを使用することができます。各ファイル・ヘッダには、ファイル識別番号が付いています。

ファイルを作成する場合、通常は、OpenVMS RMS に対してファイル名を指定します。このファイル名は、Files-11 ボリューム上のファイルに割り当てられます。OpenVMS RMS は、新たに作成したファイルのファイル名とファイル識別子を、ファイルの記憶位置を示すエントリが入っているディレクトリに入れます。このファイルにアクセスするには、ファイル名を入力します。ファイル名は、ディレクトリ・エントリを通じてファイル識別子を指すパスとなります。ファイル識別子は、ファイル・ヘッダの記憶位置を指し、ファイル・ヘッダに入っているエクステント・リストによって、実データの記憶位置が分かります。

ファイル・ヘッダは、ファイルの現在の記憶状態を示すので、ANALYZE/DISK_STRUCTURE においては特に意味を持ちます。Files-11 ディスク上のファイルは、INDEXF.SYS の 1 次ヘッダで表されます。必要に応じて、拡張ヘッダも使用されます。

各固定長ヘッダには、固定長データと可変長データの両方が入っています。このデータは、表 A-3 「ファイル・ヘッダのデータ・エリア」に示す 6 つのエリアのいずれかに入っています。

表 A-3 ファイル・ヘッダのデータ・エリア

データ・エリア	説明
Header	ヘッダ識別子、ファイル番号、ファイル・シーケンス番号、ファイルの保護コード、他のファイル・ヘッダ・エリアへのオフセットが入っている。
Ident	ファイルの識別子と会計情報データが入っている。具体的には、ファイル名、作成日時、バックアップ日時など。
Map	ファイルの仮想ブロックをボリュームの論理ブロックにマップする検索ポイントのリストが入っている。各ポイントは、ファイルに割り当てられている、連続番号が付いたブロックで構成されるグループを示す。検索ポイントの順序は、対応する仮想ブロックの順序となる。
Access control list	ACL 関係情報が入っている省略可能エリア。
Reserved	特殊なアプリケーション用に予約されている。
End checksum	ファイル・ヘッダ末尾の 2 バイトは、その前の 255 ワードのチェックサムである。チェックサムにより、正しいファイル・ヘッダであるかどうかをチェックできる。

連続するクラスタの集合を、**エクステント** と呼びます。エクステントのサイズは、連続クラスタの数によって決まります。たとえば、1000 ブロックの容量を要するファイルがあるときに、800 ブロックと 200 ブロックの連続空間が存在する場合、このファイルは 800 ブロックと 200 ブロックの 2 つのエクステントに入れられます。

ファイルの **1 次ヘッダ** は、そのファイルの第 1 エクステントをはじめ、1 次ヘッダのマップ・エリアに入れることができる数のエクステントを指します。ファイルに必要なエクステントの数がマップ・エリアを超える場合や、ACL が 1 次ヘッダに入りきらないほど大きい場合、**拡張ヘッダ** が割り当てられます。拡張ヘッダには、1 次ヘッダの固定データのほか、拡張ヘッダが指すエクステントの記憶位置を指定する可変データ (ヘッダ・マップ・エリアとアクセス制御リスト) が入ります。

ANALYZE/DISK_STRUCTURE は、1 次ヘッダと拡張ヘッダのリスト全体に対し、ファイルの妥当性をチェックします。チェック対象項目は、ファイル・ヘッダ、拡張ヘッダを指す全ポイントで構成されるチェーン、全ヘッダの検索ポイント、ファイルの属性です。

A.3.2 ストレージ・ビットマップ・ファイル BITMAP.SYS

ストレージ・ビットマップ・ファイルは、ボリューム上で使用できる空間量をファイル・システムが記録するために使用する連続ファイルです。このファイルには、ストレージ制御ブロック (SCB) が入っています。SCB には、Files-11 空間割り当てを最適化するための情報と個々のブロックの使用可能性を示すビットマップが入っています。

SCB の情報の内容は、クラスタの要素の数、ボリューム・サイズ、ブロッキング・ファクタなどです。ビットマップの各ビットは、各クラスタを示します。ビットが設定されている場合、対応するクラスタを使用することができます。ビットがクリアされている場合、クラスタを使用することはできません。

オペレーティング・システムは、ビットマップの一部をキャッシュ・メモリとの間で移動します。メモリ内の各ビットの状態は、クラスタを割り当てたり割り当て解除したりするたびに、変更されます。キャッシュに入っているビットマップをディスクに戻したとき、BITMAP.SYS は更新されます。ビットマップの一部は必ずキャッシュに入っているため、(ディスクをディスマウントするか、またはライト・ロックしないかぎり) ディスクに割り当てられているクラスタの現在の状態を BITMAP.SYS が反映することはあり得ません。

ANALYZE/DISK_STRUCTURE には、INDEXF.SYS から取り出したデータをもとに BITMAP.SYS の現在のバージョンを作成し、ディスク上の空きクラスタの状態を BITMAP.SYS に正確に反映させるという機能があります。

A.3.3 不良ブロック・ファイル BADBLK.SYS

不良ブロック・ファイルには、ボリューム上の不良ブロックがすべて入ります。システムは不良ブロックを動的に検出し、不良ブロックを使用しているファイルを削除した後にこれらの不良ブロックが再び使用されることを防止します。

A.3.4 マスタ・ファイル・ディレクトリ

MFD は、Files-11 ボリューム・ディレクトリ構造を制御する予約ファイルを含むファイルです。また MFD は、ユーザが使用するファイルやディレクトリ、および既知ファイルをリストします。マスタ・ファイル・ディレクトリ自体は、MFD でリストされるファイル (000000.DIR;1) の 1 つです。

ただし、MFD は、予約ファイルとユーザのファイル・ディレクトリのリストに使用されることが多く、プライベート・ボリュームにおいてさえ、ユーザが MFD にファイルを入力することはほとんどありません。プライベート・ボリュームでは、システム・ディスクの省略時ディレクトリと同じ名前のディレクトリを作成した方が便利です。ユーザのファイル・ディレクトリとファイル指定については、『OpenVMS ユーザーズ・マニュアル』を参照してください。

BACKUP ユーティリティで順編成ディスク・セーブ・セットを作成すると、セーブ・セット・ファイルが MFD に格納されます。

ANALYZE/DISK_STRUCTURE は、INDEXF.SYS との比較を行うことにより、ディレクトリ構造に属する全ファイルをチェックします。ディレクトリ構造で追跡できないファイルは、"失われた" ファイルです。/REPAIR が指定されている場合、これらのファイルは最上位レベルの SYSLOST.DIR ディレクトリに入れられます。

A.3.5 コア・イメージ・ファイル CORIMG.SYS

オペレーティング・システムでは、コア・イメージ・ファイルを使用していません。

A.3.6 ボリューム・セット・リスト・ファイル VOLSET.SYS

ボリューム・セット・リスト・ファイルは、ボリューム・セットの相対ボリューム 1 でのみ使用します。このファイルには、ボリューム・セットに属する全ボリュームのラベルとボリューム・セット名が入っています。

ANALYZE/DISK_STRUCTURE では、VOLSET.SYS を使用してボリューム・セット内の各ボリュームの記憶位置を調べ、各ボリュームの属性を確認します。すべてのボリューム・セット情報が相対ボリューム 1 の VOLSET.SYS に入っているため、他のボリュームに存在する VOLSET.SYS は無視されます。

A.3.7 継続ファイル CONTIN.SYS

継続ファイルは、1 つのファイルが 2 つのボリュームにまたがるときに拡張ファイル識別子として使用されます。このファイルは、順編成ディスク・セーブ・セットの最初のボリュームを除くすべてのボリュームに使用されます。

A.3.8 バックアップ・ログ・ファイル BACKUP.SYS

バックアップ・ログ・ファイルは、今後の使用のため予約されています。

A.3.9 保留不良ブロック・ログ・ファイル BADLOG.SYS

保留不良ブロック・ログ・ファイルには、不良ブロック・ファイルに入っていないけれども不良であると思われるブロックのリストが入っています。

A.3.10 クォータ・ファイル QUOTA.SYS

クォータ・ファイルは、ボリューム上の各 UIC のディスク使用量を記録するためにファイル・システムが使用する、予約ファイルです。ボリュームのクォータ・チェックを許可している場合、ボリューム上の全 UIC が QUOTA.SYS ファイルに格納されます。QUOTA.SYS は常に更新されるので、現在のディスク使用量、許可されている最大ディスク使用量、許可されている超過値が、UIC ごとに示されます。

ANALYZE/DISK_STRUCTURE は、その動作時に、各 UIC の実際のディスク使用量を反映する QUOTA.SYS のコピーをメモリに作成します。このコピーは、ディスク上の QUOTA.SYS と比較されます。矛盾点がある場合は、メッセージが表示されます。/REPAIR 修飾子を指定した場合、ディスク上の QUOTA.SYS が更新されます。

A.3.11 ボリューム機密保護プロファイル SECURITY.SYS

ボリューム機密保護プロファイルには、ボリュームの所有者 UIC、ボリュームのシステム - 所有者 - グループ - 一般ユーザ (SOGW) 保護マスク、およびボリュームのアクセス制御リスト (ACL) が含まれます。

A.4 Files-11 ODS レベル 1 と 2,5 の違い (VAX のみ)

VAX システムでは、性能、信頼性、機密保護の点で、ODS レベル 1 の互換スーパーセットである Files-11 ODS レベル 2 が標準のディスク構造です。ボリューム初期化時の省略時の値は、構造レベル 2 です。『OpenVMS DCL ディクショナリ』の INITIALIZE コマンドを参照してください。

RSX-11M, RSX-11D, RSX-11M-PLUS, IAS は ODS レベル 1 以外をサポートしていないので、これらのシステムに移植する必要がある VAX ボリュームの場合には、ODS レベル 1 を指定します。また、これらのシステムから移植した構造レベル 1 ボリュームを扱う必要が生じる場合もあります。

構造レベル 1 のボリュームを使用している場合、表 A-4 「Files-11 構造レベル 1 のボリュームにおける制限」に示す制限に注意してください。

表 A-4 Files-11 構造レベル 1 のボリュームにおける制限

ディスク	保護対象オブジェクトは、Files-11 ODS-2 ディスクだけである。
ディレクトリ	ディレクトリとサブディレクトリの階層がなく、ディレクトリ・エントリつまりファイル名の順序も定められていない。RSX-11M, RSX-11D, RSX-11M-PLUS, IAS は、サブディレクトリをサポートしておらず、ディレクトリ・エントリをアルファベット順に並べない。
ディスク・クォータ	サポートしていない。
マルチボリューム・ファイルとボリューム・セット	サポートしていない。
位置制御	サポートしていない。
キャッシュ	ファイル・ヘッダ・ブロック、ファイル識別スロット、エクステント・エントリのキャッシングを行わない。
システム・ディスク	構造レベル 1 ボリュームを使用できない。
OpenVMS Cluster アクセス	ローカル・アクセスのみ。クラスタ全体で共用できない。
クラスタ化割り当て	サポートしていない。
バックアップ・ホーム・ブロック	サポートしていない。
保護コード E	E は、RSX-11M オペレーティング・システムで "拡張" を意味するが、OpenVMS では無視される。
ファイル・バージョン	32,767 までに制限される。バージョン制限はサポートしていない。
高度保護機能(アクセス制御リストなど)	サポートしていない。
ロング・ファイル名	サポートしていない。
RMS ジャーナル機能	サポートしていない。
RMS 実行統計監視	サポートしていない。

OpenVMS ソフトウェアの今後の機能強化では、構造レベル 5 が中心となります。したがって、構造レベル 1 における制限はさらに多くなる可能性があります。

付録B 時差係数 (TDF) 表

以下の表は、世界各地のTDFを示しています。各表は、世界の特定地域の所在地リストです。なお、表中の値は、このドキュメントの刊行時において正確であると考えられる値です。



注意:

タイム・ゾーン規則は、各国が管理しており、政情およびその他の理由により変更される場合があります。最新の情報については、次のウェブ・サイトを参照してください。

<http://swissinfo.net/cgi/worldtime/> <http://times.clari.net.au/>

表 B-1 「欧州の TDF」 は、欧州の時差係数を示しています。

表 B-1 欧州の TDF

地域	標準時 TDF	夏時間 TDF
英国, アイルランド	0:00	+1:00
西欧州	0:00	+1:00
アイスランド	0:00	—
中部欧州	+1:00	+2:00
ポーランド	+2:00	+3:00
東欧州	+2:00	+3:00
トルコ	+2:00	+3:00

表 B-2 「北米の TDF」 は、北米の時差係数を示しています。

表 B-2 北米の TDF

地域	標準時 TDF	夏時間 TDF
U.S./ 東部	-5:00	-4:00
U.S./ 中部	-6:00	-5:00
U.S./ 山地	-7:00	-6:00
U.S./ 太平洋	-8:00	-7:00
U.S./ インディアナ (東部)	-5:00	—
U.S./ アラスカ	-9:00	-8:00
U.S./ アリゾナ	-7:00	—
U.S./ ナバホ	-7:00	-6:00
U.S./ ミシガン	-5:00	-4:00
U.S./ アリユーション列島	-10:00	-9:00
U.S./ ハワイ	-10:00	—
U.S./ サモア	-11:00	—
カナダ/ニューファンドランド	-3:30	-2:30
カナダ/大西洋	-4:00	-3:00
カナダ/東部	-5:00	-4:00
カナダ/中部	-6:00	-5:00
カナダ/東サスカチュワン	-6:00	—

表 B-2 北米の TDF (続き)

地域	標準時 TDF	夏時間 TDF
カナダ/山地	-7:00	-6:00
カナダ/太平洋	-8:00	-7:00
カナダ/ユーコン	-9:00	-8:00

表 B-3 「中米および南米の TDF」 は、中米および南米の時差係数を示しています。

表 B-3 中米および南米の TDF

地域	標準時 TDF	夏時間 TDF
メキシコ/バハノルチ	-8:00	-7:00
メキシコ/バハスル	-7:00	—
メキシコ/その他全土	-6:00	—
キューバ	-5:00	-4:00
ジャマイカ	-5:00	-4:00
ブラジル/東部	-3:00	-2:00
ブラジル/西部	-4:00	-3:00
ブラジル/アクレ	-5:00	-4:00
ブラジル/デノローニャ	-2:00	-1:00
チリ/その他全土	-4:00	-3:00
チリ/イースター島	-6:00	-5:00

表 B-4 「アジアの TDF」 は、アジアの時差係数を示しています。

表 B-4 アジアの TDF

地域	標準時 TDF	夏時間 TDF
PRC (中国本土)	+8:00	+9:00
ROK (韓国)	+9:00	+10:00
イスラエル	+2:00	+3:00
イラン	+3:30	+4:30
日本	+9:00	—
シンガポール	+8:00	—
香港	+8:00	—
ROC (台湾)	+8:00	—

表 B-5 「南太平洋の TDF」 は、南太平洋の時差係数を示しています。

表 B-5 南太平洋の TDF

地域	標準時 TDF	夏時間 TDF
オーストラリア/タスマニア	+10:00	+11:00
オーストラリア/クィーンズランド (標準時のみ)	+10:00	—
オーストラリア/クィーンズランド	+10:00	+11:00

表 B-5 南太平洋の TDF (続き)

地域	標準時 TDF	夏時間 TDF
オーストラリア/北部	+9:30	—
オーストラリア/西部	+8:00	—
オーストラリア/南部	+9:30	+10:30
オーストラリア/ビクトリア	+10:00	+11:00
オーストラリア/ニューサウスウェールズ	+10:00	+11:00
ニュージーランド	+12:00	+13:00

表 B-6 「南極の TDF」 は、南極の時差係数を示しています。

表 B-6 南極の TDF

地域	標準時 TDF	夏時間 TDF
南極	+0:00	—

付録C OpenVMS に用意されているタイムゾーン

OpenVMS バージョン 8.2 には、合計で 508 個のタイムゾーンがあります。このバージョンでは、204 個の新しいタイムゾーンが追加され、既存のタイムゾーンの一部がアップデートされています。変更は、次の URL で入手可能なタイムゾーン・パブリック・データベース `tzdata2003e.tar.gz` をベースにしています。

`ftp://elsie.nci.nih.gov/pub/`

この付録では、すべてのタイムゾーンをアルファベット順で記載します。

Africa/Abidjan
Africa/Accra
Africa/Addis_Ababa
Africa/Algiers
Africa/Asmera
Africa/Bamako
Africa/Bangui
Africa/Banjul
Africa/Bissau
Africa/Blantyre
Africa/Brazzaville
Africa/Bujumbura
Africa/Cairo
Africa/Casablanca
Africa/Ceuta
Africa/Conakry
Africa/Dakar
Africa/Dar_es_Salaam
Africa/Djibouti
Africa/Douala
Africa/El_Aaiun
Africa/Freetown
Africa/Gaborone
Africa/Harare
Africa/Johannesburg
Africa/Kampala
Africa/Khartoum
Africa/Kigali
Africa/Kinshasa
Africa/Lagos
Africa/Libreville
Africa/Lome
Africa/Luanda
Africa/Lubumbashi
Africa/Lusaka
Africa/Malabo
Africa/Maputo
Africa/MaseruAfrica/Mbabane
Africa/Mogadishu
Africa/Monrovia
Africa/Nairobi
Africa/Ndjamena
Africa/Niamey
Africa/Nouakchott
Africa/Ouagadougou
Africa/Porto-Novo
Africa/Sao_Tome
Africa/Timbuktu
Africa/Tripoli
Africa/Tunis
Africa/Windhoek

America/Adak
America/Anchorage
America/Anguilla
America/Antigua
America/Araguaina
America/Aruba
America/Asuncion
America/Atka
America/Bahia
America/Barbados
America/Belem
America/Belize
America/Boa_Vista
America/Bogota
America/Boise
America/Buenos_Aires
America/Cambridge_Bay
America/Campo_Grande
America/Cancun
America/Caracas
America/Catamarca
America/Cayenne
America/Cayman
America/Chicago
America/Chihuahua
America/Cordoba
America/Costa_Rica
America/Cuiaba
America/Curacao
America/Danmarkshavn
America/Dawson
America/Dawson_Creek
America/Denver
America/Detroit
America/Dominica
America/Edmonton
America/Eirunepe
America/El_Salvador
America/Ensenada
America/Fort_Wayne
America/Fortaleza
America/Glace_Bay
America/Godthab
America/Goose_Bay
America/Grand_Turk
America/Grenada
America/Guadeloupe
America/Guatemala
America/Guayaquil
America/Guyana
America/Halifax
America/Havana
America/Hermosillo
America/Indiana/Indianapolis
America/Indiana/Knox
America/Indiana/Marengo
America/Indiana/Vevay
America/Indianapolis
America/Inuvik
America/Iqaluit
America/Jamaica
America/Jujuy
America/Juneau

America/Kentucky/Louisville
America/Kentucky/Monticello
America/Knox_IN
America/La_Paz
America/Lima
America/Los_Angeles
America/Louisville
America/Maceio
America/Managua
America/Manaus
America/Martinique
America/Mazatlan
America/Mendoza
America/Menominee
America/Merida
America/Mexico_City
America/Miquelon
America/Monterrey
America/Montevideo
America/Montreal
America/Montserrat
America/Nassau
America/New_York
America/Nipigon
America/Nome
America/Noronha
America/North_Dakota/Center
America/Panama
America/Pangnirtung
America/Paramaribo
America/Phoenix
America/Port-au-Prince
America/Port_of_Spain
America/Porto_Acre
America/Porto_Velho
America/Puerto_Rico
America/Rainy_River
America/Rankin_Inlet
America/Recife
America/Regina
America/Rio_Branco
America/Rosario
America/Santiago
America/Santo_Domingo
America/Sao_Paulo
America/Scoresbysund
America/Shiprock
America/St_Johns
America/St_Kitts
America/St_Lucia
America/St_Thomas
America/St_Vincent
America/Swift_Current
America/Tegucigalpa
America/Thule
America/Thunder_Bay
America/Tijuana
America/Toronto
America/Tortola
America/Vancouver
America/Virgin
America/Whitehorse
America/Winnipeg
America/Yakutat
America/Yellowknife

Antarctica/Casey
Antarctica/Davis
Antarctica/DumontDUrville
Antarctica/Mawson
Antarctica/McMurdo
Antarctica/Palmer
Antarctica/Rothera
Antarctica/South_Pole
Antarctica/Syowa
Antarctica/Vostok

Arctic/Longyearbyen

Asia/Aden
Asia/Almaty
Asia/Amman
Asia/Anadyr
Asia/Aqtau
Asia/Aqtobe
Asia/Ashgabat
Asia/Ashkhabad
Asia/Baghdad
Asia/Bahrain
Asia/Baku
Asia/Bangkok
Asia/Beirut
Asia/Bishkek
Asia/Brunei
Asia/Calcutta
Asia/Choibalsan
Asia/Chongqing
Asia/Chungking
Asia/Colombo
Asia/Dacca
Asia/Damascus
Asia/Dhaka
Asia/Dili
Asia/Dubai
Asia/Dushanbe
Asia/Gaza
Asia/Harbin
Asia/Hong_Kong
Asia/Hovd
Asia/Irkutsk
Asia/Istanbul
Asia/Jakarta
Asia/Jayapura
Asia/Jerusalem
Asia/Kabul
Asia/Kamchatka
Asia/Karachi
Asia/Kashgar
Asia/Katmandu
Asia/Krasnoyarsk
Asia/Kuala_Lumpur
Asia/Kuching
Asia/Kuwait
Asia/Macao
Asia/Macau
Asia/Magadan
Asia/Makassar
Asia/Manila
Asia/Muscat
Asia/Nicosia

Asia/Novosibirsk
Asia/Omsk
Asia/Oral
Asia/Phnom_Penh
Asia/Pontianak
Asia/Pyongyang
Asia/Qatar
Asia/Qyzylorda
Asia/Rangoon
Asia/Riyadh
Asia/Riyadh87
Asia/Riyadh88
Asia/Riyadh89
Asia/Saigon
Asia/Sakhalin
Asia/Samarkand
Asia/Seoul
Asia/Shanghai
Asia/Singapore
Asia/Taipei
Asia/Tashkent
Asia/Tbilisi
Asia/Tehran
Asia/Tel_Aviv
Asia/Thimbu
Asia/Thimphu
Asia/Tokyo
Asia/Ujung_Pandang
Asia/Ulaanbaatar
Asia/Ulan_Bator
Asia/Urumqi
Asia/Vientiane
Asia/Vladivostok
Asia/Yakutsk
Asia/Yekaterinburg
Asia/Yerevan

Atlantic/Azores
Atlantic/Bermuda
Atlantic/Canary
Atlantic/Cape_Verde
Atlantic/Faeroe
Atlantic/Jan_Mayen
Atlantic/Madeira
Atlantic/Reykjavik
Atlantic/South_Georgia
Atlantic/St_Helena
Atlantic/Stanley

Australia/ACT
Australia/Adelaide
Australia/Brisbane
Australia/Broken_Hill
Australia/Canberra
Australia/Darwin
Australia/Hobart
Australia/LHI
Australia/Lindeman
Australia/Lord_Howe
Australia/Melbourne
Australia/NSW
Australia/North
Australia/Perth
Australia/Queensland
Australia/South

Australia/Sydney
Australia/Tasmania
Australia/Victoria
Australia/West
Australia/Yancowinna

Brazil/Acre
Brazil/DeNoronha
Brazil/East
Brazil/West

CET
CST6CDT

Canada/Atlantic
Canada/Central
Canada/East-Saskatchewan
Canada/Eastern
Canada/Mountain
Canada/Newfoundland
Canada/Pacific
Canada/Saskatchewan
Canada/Yukon

Chile/Continental
Chile/EasterIsland

Cuba

EET
EST
EST5EDT
Egypt
Eire

Etc/GMT
Etc/GMT+0
Etc/GMT+1
Etc/GMT+10
Etc/GMT+11
Etc/GMT+12
Etc/GMT+2
Etc/GMT+3
Etc/GMT+4
Etc/GMT+5
Etc/GMT+6
Etc/GMT+7
Etc/GMT+8
Etc/GMT+9
Etc/GMT-0
Etc/GMT-1
Etc/GMT-10
Etc/GMT-11
Etc/GMT-12
Etc/GMT-13
Etc/GMT-14
Etc/GMT-2
Etc/GMT-3
Etc/GMT-4
Etc/GMT-5
Etc/GMT-6
Etc/GMT-7
Etc/GMT-8
Etc/GMT-9
Etc/GMT0

Etc/Greenwich
Etc/UCT
Etc/UTC
Etc/Universal
Etc/Zulu

Europe/Amsterdam
Europe/Andorra
Europe/Athens
Europe/Belfast
Europe/Belgrade
Europe/Berlin
Europe/Bratislava
Europe/Brussels
Europe/Bucharest
Europe/Budapest
Europe/Chisinau
Europe/Copenhagen
Europe/Dublin
Europe/Gibraltar
Europe/Helsinki
Europe/Istanbul
Europe/Kaliningrad
Europe/Kiev
Europe/Lisbon
Europe/Ljubljana
Europe/London
Europe/Luxembourg
Europe/Madrid
Europe/Malta
Europe/Minsk
Europe/Monaco
Europe/Moscow
Europe/Nicosia
Europe/Oslo
Europe/Paris
Europe/Prague
Europe/Riga
Europe/Rome
Europe/Samara
Europe/San_Marino
Europe/Sarajevo
Europe/Simferopol
Europe/Skopje
Europe/Sofia
Europe/Stockholm
Europe/Tallinn
Europe/Tirane
Europe/Tiraspol
Europe/Uzhgorod
Europe/Vaduz
Europe/Vatican
Europe/Vienna
Europe/Vilnius
Europe/Warsaw
Europe/Zagreb
Europe/Zaporozhye
Europe/Zurich

Factory

GB
GB-Eire

GMT

GMT+0
GMT-0
GMT0

Greenwich

HST

Hongkong

Iceland

Indian/Antananarivo
Indian/Chagos
Indian/Christmas
Indian/Cocos
Indian/Comoro
Indian/Kerguelen
Indian/Mahe
Indian/Maldives
Indian/Mauritius
Indian/Mayotte
Indian/Reunion

Iran

Israel

Jamaica

Japan

Kwajalein

Libya

MET
MST
MST7MDT

Mexico/BajaNorte
Mexico/BajaSur
Mexico/General

Mideast/Riyadh87
Mideast/Riyadh88
Mideast/Riyadh89

NZ
NZ-CHAT

Navajo
PRC
PST8PDT

Pacific/Apia
Pacific/Auckland
Pacific/Chatham
Pacific/Easter
Pacific/Efate
Pacific/Enderbury
Pacific/Fakaofo
Pacific/Fiji
Pacific/Funafuti

Pacific/Galapagos
Pacific/Gambier
Pacific/Guadalcanal
Pacific/Guam
Pacific/Honolulu
Pacific/Johnston
Pacific/Kiritimati
Pacific/Kosrae
Pacific/Kwajalein
Pacific/Majuro
Pacific/Marquesas
Pacific/Midway
Pacific/Nauru
Pacific/Niue
Pacific/Norfolk
Pacific/Noumea
Pacific/Pago_Pago
Pacific/Palau
Pacific/Pitcairn
Pacific/Ponape
Pacific/Port_Moresby
Pacific/Rarotonga
Pacific/Saipan
Pacific/Samoa
Pacific/Tahiti
Pacific/Tarawa
Pacific/Tongatapu
Pacific/Truk
Pacific/Wake
Pacific/Wallis
Pacific/Yap

Poland
Portugal
ROC
ROK
Singapore

SystemV/AST4
SystemV/AST4ADT
SystemV/CST6
SystemV/CST6CDT
SystemV/EST5
SystemV/EST5EDT
SystemV/HST10
SystemV/MST7
SystemV/MST7MDT
SystemV/PST8
SystemV/PST8PDT
SystemV/YST9
SystemV/YST9YDT

Turkey

UCT

US/Alaska
US/Aleutian
US/Arizona
US/Central
US/East-Indiana
US/Eastern
US/Hawaii
US/Indiana-Starke
US/Michigan

US/Mountain
US/Pacific
US/Pacific-New
US/Samoa

UTC
Universal
W-SU
WET
Zulu

用語集

この用語集は、『OpenVMS システム管理者マニュアル』で使用した用語とその定義をアルファベット順および五十音順で並べています。

1 次ブートストラップ・イメージ	ブート・ブロックが示すプログラム。このプログラムは 2 次ブートストラップ・イメージ SYSBOOT.EXE を検索してシステム・ディスクへのアクセスを可能にし、このイメージをメモリにロードする。 VAX システムの場合、1 次ブートストラップ・イメージは VMB.EXE である。 Alpha システムの場合、1 次ブートストラップ・イメージは APB.EXE である。
1 次プロセッサ	マルチプロセッシング・システムにおいて、論理的または物理的にコンソール・デバイスに接続され、マルチプロセッシング・システムをブートするコンソール・コマンドの実行対象であるプロセッサ。1 次プロセッサは、マルチプロセッシング・システムに存在する他のプロセッサを起動する。システム・タイムキーバも担当する。
1 次ページ・ファイルおよび 1 次スワップ・ファイル	ディストリビューション・キットに入っている、省略時の ページ・ファイル と スワップ・ファイル 。ファイル名は、SYS\$SYSTEM:PAGEFILE.SYS と SYS\$SYSTEM:SWAPFILE.SYS である。 2 次ページ・ファイル および 2 次スワップ・ファイル を参照。
2 次ブートストラップ・イメージ	システム・ディスクへのアクセスを可能にするイメージ。SYS\$SYSTEM:SYSBOOT.EXE。
2 次プロセッサ	マルチプロセッシング・システムにおいて、 1 次プロセッサ ではないプロセッサ。
2 次ページ・ファイルおよび 2 次スワップ・ファイル	性能やディスク空間の都合で付加的に作成される ページ・ファイル と スワップ・ファイル 。 1 次ページ・ファイル と 1 次スワップ・ファイル の空間に加え、これらの 2 次ファイルの空間がページングとスワッピングに使用される。
ACL	アクセス制御リスト を参照。
BOT マーカ	テープ・ボリュームの書き込み可能領域の始まりを示す光反射性のマーク。
CD-ROM	オーディオ機器で使用される CD-ROM と同様のコンピュータ用ディスク記憶デバイス。オーディオ用 CD-ROM との相違は、CD-ROM コンピュータ・プレイヤはオーディオ・インタフェースではなく、デジタル・インタフェースを使用する点である。
CMIP	Common Management Information Protocol(CMIP) を参照。
Common Management Information Protocol(CMIP)	DECnet-Plus ネットワーク管理プロトコルに基づいた、ネットワーク管理操作の標準案。CMIP を使用して、要素で実行されるネットワーク管理操作をコーディングする。CMIP は、ディレクタとエージェントとの情報の交換を許可する。また、CMIP を使用すると、Phase IV Network Information および Control Exchange(NICE) プロトコルは使用できない。
Coordinated Universal Time (UTC)	1 日の時刻を表すための国際標準規約。
DECnet	弊社のオペレーティング・システムをネットワークに接続するためのハードウェアおよびソフトウェアの汎用名。DECnet を使用すると、1 つのシステムは 1 つのノードのように機能する。 DECnet Phase IV 、 DECnet-Plus for OpenVMS 、および TCP/IP Services for OpenVMS を参照。
DECnet Phase IV	DECnet の古いバージョン。レイヤード製品のように、OpenVMS とは別にインストールして使用する。
DECnet-Plus for OpenVMS	DECnet の新しいバージョン。OpenVMS オペレーティング・システムのインストレーション・プロシージャを実行すると使用できる。DECnet-Plus は、Digital Network Architecture(DNA) Phase V の弊社のインプリメントです。
EOT マーカ	テープ・ボリュームの書き込み可能領域の終端を表す光反射性のマーク。
ERRFMT プロセス	システム・プロセスの 1 つ。定期的にエラー・ログ・バッファを空にし、エラーの記述を標準書式に変換してシステム・ディスク上の エラー・ログ・ファイル に格納する。
Files-11 ディスク構造	ディスク上に格納される情報に与えられる論理構造。ファイルとそのデータ、およびそれらのデータにアクセスするためのディレクトリを階層的に編成したもの。

Files-11 ボリューム	Files-11 ディスク構造を使用し、デバイスにマウントされるディスク・ボリューム。
InfoServer システム	イーサネットに基づいた高性能 仮想デバイス・サーバ 。InfoServer システムにより、ローカル・エリア・ネットワーク (LAN) 内のクライアント・システムは、物理デバイス媒体や論理ディスク・ブロックのセットを使用することができるようになる。適切なクライアント・ソフトウェアを実行しているシステムは、InfoServer システムの仮想デバイスに接続し、ローカルに接続されたデバイスと同様にそれらを使用することができる。
LASTport プロトコル	InfoServer ソフトウェアによりインプリメントされる 特別な LAN トランスポート・プロトコル。これにより、多くのクライアントが InfoServer システムにアクセスし、信頼性の高いデバイスの読み込みおよび書き込み操作を行うことができる。 LASTport/DISK プロトコルと LASTport/TAPE プロトコルは、LASTport プロトコルを使用するディスクおよびテープの特別なプロトコルである。 InfoServer システム も参照。
LAT サービス	LAT ソフトウェアによって LAN 上のユーザに使用可能なシステム資源。 汎用タイムシェアリング・サービス および アプリケーション・サービス を LAT サービスにすることができる。
LAT サービス・ノード	外部からの LAT 接続をサポートするシステム、または LAT サービスを提供するシステム。
LAT サービス通知	LAT サービス・ノード によって送信されたマルチキャスト・メッセージ。このメッセージによって、使用可能なサービス・ノードのデータベースが構築される。
LAT プロトコル	LAT ソフトウェアによって実現されるプロトコル。このプロトコルにより、オペレーティング・システムは資源、すなわち ターミナル・サーバ がアクセスできる LAT サービスを提供する。
MSCP サーバ	OpenVMS Cluster 環境において、MSCP プロトコルを実現するコンポーネント。RA シリズ・ディスクのような DSA ディスク用コントローラとの通信に使用される。ディスク・クラスの デバイス・ドライバ である DUDRIVER および DSDRIVER の一方または両方と組み合わせることで、MSCP サーバはコンピュータ上で MSCP プロトコルを実現し、そのコンピュータが記憶域制御デバイスとして機能することを可能にする。
OPCOM プロセス	オペレータ通信マネージャ (OPCOM) の動作を管理するシステム・プロセス。
OPCOM メッセージ	オペレータ通信マネージャ (OPCOM) がブロードキャストするメッセージ。これらのメッセージは オペレータ・ターミナル に表示され、 オペレータ・ログ・ファイル に書き込まれる。システム管理者が送信する一般的なメッセージ、ユーザからの要求、オペレータの応答、システム・イベントなどのメッセージがある。
OpenVMS Cluster システム	複数のコンピュータと記憶サブシステムの疎結合。VMScluster システムは、システム資源の一部または全部を共有するが、ユーザからは単一システムのように見える。複数のコンピュータが VMScluster 環境の資源を共有すると、すべてのコンピュータの記憶資源とコンピューティング資源が組み合わせられるので、処理能力が向上する。
PAK	製品登録キー (PAK) を参照。
SCS	OpenVMS Cluster 環境において、HP SCA (システム通信アーキテクチャ) に基づいてコンピュータ間通信を実現するソフトウェア。
SYSGEN パラメータ	システム・パラメータ を参照。
TCP/IP Services for OpenVMS	弊社の TCP/IP プロトコル、および OpenVMS Alpha, I64, および VAX オペレーティング・システムへのインターネット・サービスのインプリメントです。
TMSCP サーバ	OpenVMS Cluster 環境において、TU シリズ・テープなどの MSCP ローカル・テープのコントローラとの通信に使用する TMSCP プロトコルを実現するコンポーネント。TMSCP サーバがテープ・クラス・ デバイス・ドライバ (TUDRIVER) とともにこのプロトコルをプロセッサで実現することにより、プロセッサが記憶デバイスのコントローラとして機能する。
UAF	ユーザ登録ファイル (UAF) を参照。
UETP	OpenVMS オペレーティング・システムが正常にインストールされたかどうかを確認するソフトウェアのこと。
UIC	ユーザ識別コード (UIC) を参照。

UIC に基づく保護	UIC (ユーザ識別コード) に基づき、保護されるすべてのオブジェクトに適用される保護メカニズム。 ACL (アクセス制御リスト) を参照。
UTC	Coordinated Universal Time を参照。
VAXcluster サーバ	ローカル・エリア VAXcluster 構成において、 MSCP(大容量記憶制御プロトコル) サーバ と TMSCP(磁気テープ大容量記憶デバイス制御プロトコル) サーバ のソフトウェアを使用する VAXcluster ノード。これにより、このノードにローカル接続されているディスクとテープを、ローカル・エリア・ネットワーク (LAN) を通じて VAXcluster サテライト から利用できる。
VAXport ドライバ	VAXcluster 環境において、ローカル・ポートと遠隔ポート間の通信パスを制御する デバイス・ドライバ 。CI では PADRIVER, LAN では PEDRIVER, DSSI では PIDRIVER である。
アカウントング情報ファイル	会計情報ファイル を参照。
アカウント	各システム・ユーザにはアカウントが割り当てられ、ログイン時には特定のアカウント名とアカウント番号が使用される。システムは、ユーザが所有するファイルの記憶位置、ユーザの各ファイルに対するアクセス権、ユーザの提供できるシステム機能をアカウント番号により識別する。
アクセス・モード	ソフトウェアの実行時は 4 種類のプロセッサ・アクセス・モードのいずれかが使用される。プロセッサ・アクセス・モードにより、システム・ソフトウェアが誤ってシステムを破壊することを防ぐことができる。プロセッサ・アクセス・モードには、特権と信頼性 (保護の度合い) の高い順に、カーネル・モード、エグゼクティブ・モード、スーパーバイザ・モード、ユーザ・モードがある。プロセッサ・モードがカーネル・モード以外の場合、プロセッサは特権付き命令の実行を禁止される。
アクセス制御リスト	UIC に基づく保護 よりも洗練された保護メカニズム。個々のユーザまたはユーザ・グループに対するアクセスの許可または禁止を定義できる。
アクティブ・セット	OpenVMS VAX のマルチプロセッシング・システムにおいて、電源投入時の診断プログラムにパスし、システムの動作にアクティブに関与しているプロセッサの集合。 使用可能セット を参照。
アクティブ値	メモリに格納され、稼働中のシステムによって使用されているシステム・パラメータ値の集合。システムはそのブート時に、ディスク上のパラメータ・ファイルに格納されている 現在値 をメモリに読み込む。その値がアクティブ値となる。
アップグレード・プロシージャ	標準バージョンのオペレーティング・システムを使用している場合、アップグレード・プロシージャを実行して上位のバージョンに移行できる。
アップデート・プロシージャ	以前のバージョンのオペレーティング・システムを多少変更したい場合に使用するプロシージャ。アップデート・プロシージャは、システム・ファイルの一部を置換する。
アプリケーション・サービス	LAT サービス の 1 つ。このサービスでは LAN ユーザは 1 つの特定のプログラムにしかアクセスできない。 汎用タイムシェアリング・サービス を参照。
アベイラブル・セット	マルチプロセッシング・システムにおいて、システムの電源投入時のハードウェア診断テストにパスしたプロセッサの集合。そのプロセッサがシステムにアクティブに関与しているかどうかは問題にされない。 アクティブ・セット を参照。
アンサー・ファイル	SYS\$UPDATE: 製品名 .ANS というファイル・タイプを持つファイル。製品の初期インストール時、VMSINSTAL.COM で Auto-Answer オプションを指定するとこのファイルが作成され、インストール中にプロンプトに対して入力した値が記録される。
イーサネット	すべてのノードに対して等しいアクセス権が与えられる単一の共用ネットワーク・チャンネル。イーサネットにより、ローカル接続および遠隔接続が 1 つのネットワークに統合される。
イベント・クラス	機密保護関連のイベントのカテゴリ。システムは常に複数のイベント・クラスを監査している。
イメージ	実行可能プログラムを形成するためにリンカ・ユーティリティによって結合されたプロシージャとデータの集合。実行可能プログラムはプロセスによって実行できる。通常、可能プログラムのファイル・タイプは .EXE である。
イメージ・コピー	BACKUP ユーティリティの動作の 1 つで、出力ディスク上に Files-11 ディスク構造を作成し、ボリューム全体をコピーするもの。イメージ・バックアップは、ディスク内容の論理的なコピーである。

イメージ・バックアップ	BACKUP ユーティリティの動作の 1 つで、ディスク (またはボリューム) 上のすべてのファイルのコピーを セーブ・セット と呼ばれる特殊ファイルに保存するもの。完全バックアップとも呼ばれる。 イメージ操作 も参照。
イメージ・リストア	BACKUP ユーティリティの出力ディスクを初期化し、ボリューム全体を復元する動作。
イメージ操作	BACKUP ユーティリティの入力ディスク上のすべてのファイルを処理する動作。
イメージ登録ファイル	イメージ登録機能に関連するファイル。オペレーティング・システムの前のバージョンに依存する互換性のあるアプリケーション・イメージを続けて使用する場合に、イメージをイメージ登録ファイルに登録することができる。
イメージ比較	BACKUP ユーティリティのボリューム全体の内容を比較する動作。
インストール・プロシージャ	オペレーティング・システムの初期インストールに使用するプロシージャ。また、レイヤード製品のインストールに使用するプロシージャ。
運用時データベース (Volatile database)	ネットワークのノードにおいて、現在のネットワーク状況を反映する DECnet 構成データベース のワーキング・コピー。 パーマネント・データベース を参照。
永久オープン・イメージ	イメージ・ファイルのディレクトリ情報がメモリに永久的に存在し、ファイルを取り出すときに通常必要となるディレクトリ検索が不要な 既知イメージ 。
エクステンツ	Files-11 ボリューム上における、特定のファイルに割り当てられた連続したブロック。
エグゼクティブ	入出力、資源割り当て、プログラムの実行を行うルーチンの実行を制御する、オペレーティング・システム内のプログラムのセット。 エグゼクティブ・ルーチン も参照。
エグゼクティブ・モード	2 番目に高い特権を持つプロセッサ・ アクセス・モード 。OpenVMS RMS (レコード管理サービス)、およびオペレーティング・システムの多くのシステム・サービス・プロシージャは、エグゼクティブ・モードで実行される。
エグゼクティブ・ルーチン	エラーおよびイベントを検出して関連する情報をメモリ内のエラー・ログ・バッファに書き込むシステム・ルーチン群。 エグゼクティブ も参照。
エラー・レポート・フォーマット (ERF)	ERROR LOG ユーティリティによって呼び出され、 エラー・ログ・ファイル から選択された項目を出力するシステム・コンポーネント。
エラー・ログ・ダンプ・ファイル	Alpha システムおよび i64 システムで、エラー・ログ・バッファの内容が書き込まれるファイル (システム・ダンプ・ファイル以外)。エラー・ログ・ダンプ・ファイルが提供されるため、システムは、作成したものの、システム・クラッシュが発生した時点でまだ書き込んでいないエラー・ログ・エントリを含むように、リポート時に更新できる。
エラー・ログ・ファイル	デバイスと CPU のエラー・メッセージは、オペレーティング・システムにより自動的にこのファイルに記録される。ERROR LOG ユーティリティは、 エラー・レポート・フォーマット (ERF) を呼び出してエラー・ログ・ファイルの内容の指定された項目を出力する。
エリア・ルータ	ネットワークにおいて、エリア間およびローカル・エリア内でルーティングを行うノード。 レベル 2 ルータ とも呼ばれる。 レベル 1 ルータ を参照。
遠隔ノード	ネットワークを通してローカル・ノードにアクセスできるノード。 SYSMAN ユーティリティでは、SYSMAN を実行しているノード以外のノード。 ローカル・ノード を参照。
エンド・ノードオブジェクト	ネットワークにおいて、ルーティング動作を行わないノードのこと。 ネットワークにおける、 論理リンク の接続先プロセス。MAIL オブジェクトなどは DECnet プログラムであり、他のオブジェクトはユーザが作成したプログラムである。 ネットワークを通して 2 つのプログラムが通信する場合、ローカル・ノード上のソース・プログラムが遠隔ノード上のオブジェクトとの間に 論理リンク を確立する。
オペレータ・ターミナル	オペレータ通信マネージャ (OPCOM) がブロードキャストしたメッセージを表示するターミナル。通常は、デバイス名 OPA0: のコンソール・ターミナルがオペレータ・ターミナルである。ただし、どのユーザ・ターミナルでもオペレータ・ターミナルとして指定できる。
オペレータ・ログ・ファイル	オペレータ通信マネージャ (OPCOM) は、このファイルにメッセージを記録する。ファイル名は、SYS\$MANAGER:OPERATOR.LOG である。

カーネル・モード	プロセッサの アクセス・モード のうち最も高い特権を持つもの。入出力ドライバやページャのような最も高い特権を持つサービスはカーネル・モードで稼働する。カーネル・モードでは、プロセッサはシステムを完全に制御できる。
会計情報ファイル	システムが資源の使用状況に関する情報を格納するファイル。 カレント会計情報ファイル を参照。
会話型ブート	ブート操作の1つ。ブートを行う前に停止して、システム・パラメータ変更などの特殊な操作を行う。 ノンストップ・ブート を参照。 このブート方法は、実験、テスト、デバッグのときに動作環境を頻繁に変更するプログラム研究開発環境で利用される。
書き込み可能イメージ	参照しているプロセスがないため、またはページング上の理由で物理メモリから削除され、イメージ・ファイルに書き戻される、参照時にコピーを行わない書き込み可能共用セクションを持つ 既知イメージ 。
仮想デバイス・ユニット	InfoServer システムにおいて、遠隔サーバ上のボリュームに対して、ローカル OpenVMS のコンテキストを表現する仮想デバイス。 仮想ディスク・ユニットの場合は、DADn: という形式のデバイス名、仮想テープ・ユニットの場合は、MADn: という形式のデバイス名を持つ。 バインド、InfoServer システム、仮想デバイス・サーバ も参照。
カテドラル・ウィンドウ	マッピング・ウィンドウから作成されたセグメント化ウィンドウ。大きいファイルを読み取る時に必要なオーバヘッドの削減に役立つ。ユーザが作成できるカテドラル・ウィンドウの数は入力バイト数制限 (BITLM) により制限される。
カレント会計情報ファイル	OpenVMS Cluster 環境では、特定のノード上の アカウント・ファイル のこと。省略時の設定では、SYS\$MANAGER:ACCOUNTING.DAT がカレント会計情報ファイルである。
完全バックアップ完全名	イメージ・バックアップ を参照。 VAX システムの場合、DECdns 命名サービスに格納できる階層構造の DECnet/OSI ノード名のこと。VAX システムの完全名の最大長は 255 バイト。
キー・プロセス	逆リンクされる遷移ページも含めて、PT, S0/S1, S2 のすぐ後にダンプされるプロセス。システム管理者はキー・プロセスとして取り扱う追加プロセスを指定できる。キー・プロセスは、ダンプで他のプロセスより優先されるため、ダンプ・ファイルが小さすぎるために、すべてのプロセスを格納できない場合でも、キー・プロセスとして選択したプロセスは正しく書き込まれる。
既知イメージ	INSTALL ユーティリティによってインストールされたイメージのこと。イメージをインストールすると、そのイメージには属性が割り当てられ、システムに認識されて「既知」のイメージになる。
既知ファイル・リスト	内部データ構造の1つ。システムはこのリストに 既知イメージ を定義する。既知ファイル・リストの各エントリには、既知イメージのファイル名とそのファイルのインストール時の属性が示される。
基本プロセス優先順位	システムがプロセスをスケジューリングするときに使用する優先順位の基準値。優先順位は最も低い値が 0、最も高い値が 31 である。レベル 0 から 15 はタイムシェアリング・プロセスの優先順位、レベル 16 から 31 はリアルタイム・プロセスの優先順位である。 ジョブ・スケジューリング優先順位 を参照。
機密保護監査ログ・ファイル	システムの機密保護イベントの記録が入っているクラスタ単位のファイル。ANALYZE/AUDIT コマンドを実行すれば、機密保護監査ログ・ファイルをもとに機密保護イベントのレポートと要約を出力できる。
キャッシング	情報をメモリ内に保持することによって性能を向上させる機能。このような情報には、ディスク・ボリュームの空き領域に関するデータ、ファイル識別情報、クォータ・ファイル・エントリ、およびファイル・ヘッダが含まれる。
キャパシティ (Capability)	VAX システムにおいて、システム・ユーザに使用可能なベクタ・プロセッサのサービスを作成するソフトウェア。
キュー	ユーザが出力処理またはバッチ処理の要求を出せるようにするもの。システムは、資源の状況に従って、ユーザのプリント・ジョブやバッチ・ジョブを処理する。
キュー・データベース	キューおよび、バッチ・ジョブとプリント・ジョブに関する情報が入っている1つ以上のファイル。

キュー・マネージャ	キューの動作を制御するシステム・コンポーネント。
キュー特性	キューで実行されるバッチ・ジョブまたはプリント・ジョブを制御するために、ユーザが定義し、キューに割り当てることができる特性。
共有資源	OpenVMS Cluster 環境において、VMScluster 内のすべてのノードがアクセスできる、ディスクまたはキューなどの資源。共有資源を持つ 1 つのクラスタに属するユーザは、データ・ファイル、アプリケーション・プログラム、プリンタがどのノードに存在しているかにかかわらず、これらの資源にアクセスできる。
共用イメージ	読み込み専用セクションと参照時のコピーを行わない読み込みまたは書き込みセクションを、複数のユーザが同時にアクセスできるため、これらのセクションの 1 つのコピーを物理メモリに入れておけばよい 既知イメージ 。
共用可能イメージ	リンカ・ユーティリティの /SHAREABLE 修飾子とリンクされるイメージ。このイメージは、使用する実行可能イメージにリンクする必要がある。共用可能イメージは、 リンク可能イメージ とも呼ばれる。
クォータ・ファイル	Files-11 ボリュームにおいて、ディスクの使用が許可されているすべてのユーザを記録し、その現在のディスク使用量と使用上限値を示すファイル。クォータ・ファイル QUOTA.SYS は、他のファイルとともに [000000] ディレクトリに保存される。このファイルのエントリ 16 個に対し、1 ブロックのディスク記憶領域が必要である。 ディスク・クォータ を参照。
クラスタ	Files-11 媒体上の論理的にまとめられたブロックのグループ。ディスク空間の割り当てを行うときの基本単位として使用される。 OpenVMS Cluster システム も参照。
クラッシュ・ダンブ	オペレーティング・システムは、回復不可能なエラーまたはシステムを異常終了させる内部的な矛盾を検出した場合、エラー・ログ・バッファ、プロセッサ・レジスタ、およびメモリの内容を システム・ダンブ・ファイル に書き込む。
クラッシュ履歴ファイル	システム・クラッシュに関する情報が収められているファイル。クラッシュ履歴ファイルの内容を表示するには、クラッシュ・ログ・ユーティリティ・エキストラクタ (CLUE) を使用する。クラッシュ履歴ファイルは、クラッシュに関する問題の理解や解決、その他の有用なデータの入手に利用することができる。
グループ・ボリューム	グループ内の全ユーザに使用可能なボリューム。 システム・ボリューム を参照。
ゲートウェイ	TCP/IP ネットワークにおいて、2 つのネットワークに接続していて、1 つのネットワークからもう 1 つのネットワークにパケットを送るコンピュータ。
現在値	ディスク上の省略時のパラメータ・ファイルに格納され、システムのブートに使用されるシステム・パラメータ値の集合。システムのブート時、現在のパラメータ値がメモリに読み込まれ、 アクティブ値 が設定される。
構成データベース	ネットワーク内の各ノードには、構成データベースがあり、ここに、そのノードおよびそのノードが通信を行うことができる他のノードに関する情報が格納されている。構成データベースは パーマネント・データベース と 運用時データベース から構成される。
公用ボリューム	システム上のすべてのユーザがアクセスすることができ、プライベート・ファイルと公用ファイルの両方を記録することができる Files-11 ボリューム。
コマンド・プロシージャ	DCLのコマンドを含むファイル。それらのコマンドが使用するデータを含む場合もある。コマンド・プロシージャを実行すると、そのファイルが読み込まれ、その中のコマンドが実行される。これにより、各コマンドを別々に入力する必要はなくなり、ルーチン作業を効果的に行うことができる。また、コマンド・プロシージャはバッチ・モードで実行できる。
コンパクト・ディスク読み込み専用メモリ	オーディオ機器で使用される CD-ROM と同様のコンピュータ用ディスク記憶デバイス。オーディオ用 CD-ROM との相違は、CD-ROM コンピュータ・プレイヤはオーディオ・インタフェースではなく、デジタル・インタフェースを使用する点である。
サーキット	ネットワーク上の 隣接ノード 同士を接続する通信データ・パスのこと。サーキットは物理的なデータ・パスではなく、むしろ物理的な接続 (回線) 上で動作する論理的な接続といえる。ノード相互間のすべての入出力動作はサーキット上で行われる。

サーバ・キュー	キューに登録されているプリント・ジョブに属するファイルを、ユーザが変更した シンビオン またはユーザが作成したシンビオンを使用して処理するタイプの出力実行キュー。 プリンタ・キュー と ターミナル・キュー を参照。
再構成	製品インストール後の、(インストール中に選択した)構成の変更。
サイジング	システム資源(メモリとディスク空間)の割り当て量と、サイトの作業負荷要求とを一致させる処理。システムを自動的にサイジングするには、AUTOGEN コマンド・プロシージャを使用する。
再設定モジュール	各プリント・ジョブの終わりに挿入される デバイス制御モジュール 。ジョブの終わりでプリンタを再設定するとき使用する。
磁気テープ大容量記憶デバイス制御プロトコル・サーバ	
	OpenVMS Cluster 環境において、TU シリーズ・テープなどの MSCP ローカル・テープのコントローラとの通信に使用する TMSCP プロトコルを実現するコンポーネント。TMSCP サーバがテープ・クラス・ デバイス・ドライバ (TUDRIVER)とともにこのプロトコルをプロセッサで実現することにより、プロセッサが記憶デバイスのコントローラとして機能する。
識別レコード	ファイル・ヘッダ 内のディスクおよびボリュームの属性を含むレコード。
時差係数 (TDF)	ローカル・システム時刻と、世界標準時 (UTC) との差。 Coordinated Universal Time 参照。
システム・イメージ	オペレーティング・システムの制御下では実行されないイメージ。スタンドアロン専用である。システム・イメージの内容と形式は、 共用可能イメージ や 実行可能イメージ とは異なる。
システム・イメージ・スナップショット	スナップショット機能で使用されるシステム設定のレコード
システム・ダンプ・ファイル	システム障害の原因となる回復不能エラーや非整合性をオペレーティング・システムが検出した場合に、エラー・ログ・バッファ、プロセッサ・レジスタ、メモリの内容が書き込まれるファイル。 クラッシュ・ダンプ を参照。
システム・ディスク	OpenVMS オペレーティング・システム・ファイルが入っているディスク。
システム・パラメータ	システムをどのように機能させるかを制御するパラメータ。システム・パラメータ値は、メモリ管理、プロセス・スケジューリング、システム・セキュリティなど、各種のシステム機能を制御する。
システム・ボリューム	システム上のすべてのユーザが使用できるボリューム。 グループ・ボリューム を参照。
システム・メッセージ	DCL またはユーティリティでコマンドを入力したときにシステムが返すメッセージ。コマンドの出力内容の意味を理解するために有用である。
システムワイド論理名	システム全体で使用される論理名。システム論理名テーブルで定義される名前であり、システム内のあらゆるプロセスで使用できる。
システム通信サービス (SCS)	OpenVMS Cluster 環境において、HP SCA (システム通信アーキテクチャ) に基づいてコンピュータ間通信を実現するソフトウェア。
システム領域	CD-ROM ボリューム空間を 2 つに分割したものの 1 つで、論理セクタ 0 から 15 までを含む。システム用に予約されている領域。
実行可能イメージ	プロセス内で実行することができるイメージ。Linker ユーティリティで /EXECUTABLE 修飾子を指定することによって (または /SHAREABLE 修飾子を指定しないことによって) リンクされる。
実行キュー	バッチ・ジョブまたはプリント・ジョブを処理するために登録するキュー。 汎用キュー を参照。
自動起動キュー	自動起動機能 の特徴を利用した実行キュー。キューの作成時、そのキューを自動起動キューとして指定できる。
自動起動機能	スタートアップを簡略化し、OpenVMS Cluster 環境内の実行キューを複数のノードから利用できるようにする機能。この機能により次のことが可能となる。 <ul style="list-style-type: none"> • 1 つのコマンドでノード上のすべての自動起動キューを起動する。 • 必要に応じてあるキューが自動的にフェールオーバーできる OpenVMS Cluster 環境内のノードのリストを指定する。

自動構成	自動構成は、システムにあるハードウェア・デバイスを検出し、それに対して適切なデバイス・ドライバをロードする処理である。
出力形式	出力形式と出力キューにより、マージンやページ長などのページ・フォーマット属性を決定できる。さらに、形式に指定した用紙ストックによって、ジョブの出力が決定される。ジョブの形式がキューにマウントされた形式のストックと異なる場合、ジョブは出力されない。 省略時の出力形式は、DEFAULT と呼ばれるものが提供される。ユーザが要件に応じたフォーマットを行えるようにするため、また、特殊な用紙を必要とするジョブを出力する場合など、この出力形式以外の形式を作成できる。
出力実行キュー	シンビオント が処理するジョブを受け付けるキュー。 キュー・マネージャ は、ジョブを発行したときにユーザが指定したファイル・リストをシンビオントに送る。出力シンビオントは、ディスクから出力デバイスにデータを転送する。シンビオントは各ファイルを処理し、それと同時に、プリンタやターミナルなど、シンビオントが制御するデバイスに対して出力する。
順編成	磁気テープ上のデータ編成。テープに書き込まれた順序でデータが並べられる。
常駐イメージ	Alpha システムまたは I64 システムにおいて、 共用可能イメージ の性能を向上させる 既知イメージ 。常駐イメージを使用すると、コードが入っているイメージ部分はシステム空間に移動され、大きな単一のページに置かれるため性能が向上する。
省略時の値	ディストリビューション・キットで提供される。省略時のリストに保存されるシステム・パラメータ値の集合。これらの値により、現在サポートされているどの構成もブートできる。
初期化ファイル	一部のユーティリティで、そのユーティリティを呼び出すたびに使用されるファイル。初期化ファイルでは、キーの定義や環境の設定などを行うことができる。
ジョブ・コントローラ	バッチ・ジョブのタスクを実行するプロセスを生成するシステム・プロセス。
ジョブ・スケジューリングの優先順位	システムがキューに登録されたバッチ・ジョブまたはプリント・ジョブをスケジューリングするときに使用する優先順位の値。ジョブ・スケジューリングの優先順位の範囲は、最も低い値が 0、最も高い値が 255 である。 基本プロセス優先順位 を参照。
ジョブ・バナー・ページ	ジョブを識別するための バナー・ページ 。各ユーザは、管理者がキューのための設定したジョブ・バナー・ページを変更できる。 ファイル・バナー・ページ を参照。
所有者 UIC (Owner UIC)	UIC に基づく保護 とともに使用する。通常は、ファイルまたはボリュームを作成したユーザの UIC である。
シリンダ	ディスクのすべての記録面上の同一半径を持つすべての トラック 。
シンビオント	出力キューとともに使用され、プリント・ジョブの書式化とプリンタへの送信を行うプロセス。 オペレーティング・システムで用意されている標準のプリント・シンビオント PRTSMB は、基本の出力デバイスにファイルを出力するときに使用する。 ターミナル・サーバ に接続されている出力デバイスにファイルをプリントする場合には、LAT プリント・シンビオント LATSMB を使用する。
スタートアップ・データベース	システム・ソフトウェアのスタートアップに使用する情報が入っているファイル。たとえば、 汎用スタートアップ・コマンド・プロシージャ では、スタック・データベース STARTUP\$STARTUP_VMS に入っている情報を使用して、オペレーティング・システムを起動する。レイヤード製品を起動する場合に使用するスタートアップ・データベースは、STARTUP\$STARTUP_LAYERED である。
スプール・プリンタ	ディスクなどの中間記憶デバイスに出力を書き込むようにプリンタを設定すること。プリント・ジョブをキューに登録するのではなく、データをプリンタに直接書き込んだりコピーしたりするアプリケーションを使用している場合は、スプール・プリンタを使用する。この結果、プログラムを実行している間も、他のシステム・ユーザがプリンタを使用できる。
スライス	Alpha システムまたは I64 システムにおいて、イメージの内容を分割し、その分割した内容を同じページ保護を持つ他の分割内容と同じメモリ領域内に置くことができるようにソートするオペレーティング・システムの機能。その結果、ロード可能なエグゼクティブ・イメージあるいは共用可能イメージを従来の方法でロードする場合よりも効率的に、Alpha システムまたは I64 システム上の変換バッファを使用することができる。

スワッピング	ディスクに入っているファイルと物理メモリとの間で情報を移動することにより、システム全体に割り当てた物理メモリを効率よく使用するメモリ管理動作。システムは、アクティブとなる頻度の低いプロセスの作業領域全体を物理メモリからファイルに移動する。 ページング を参照。
スワップ・ファイル	スワッピング 動作において、スワップしたメモリ部分をシステムが書き込むファイル。ディストリビューション・キットには、SYS\$SYSTEM:SWAPFILE.SYS と名付けられたスワップ・ファイルが入っている。
製品記述ファイル (PDF)	ソフトウェア・メーカーが提供するファイルであり、ソフトウェア製品やソフトウェア製品群をインストールするために POLYCENTER Software Installation ユーティリティが必要とするすべての情報を格納する。
製品構成ファイル (PCF)	オプションとして提供される POLYCENTER Software Installation ユーティリティ・ファイルであり、ソフトウェア・メーカーから提供されるか、またはシステム管理者が作成できる。PCF には、製品のインストールで出力される質問の一部または全部に対する応答を格納する。省略時の設定や必須設定を指定することができ、これらの設定は、PDF で提供される省略時の設定と異なってもかまわない。
製品データベース (PDB)	POLYCENTER Software Installation ユーティリティによって自動的に作成されるデータベース。製品をインストールするときに、ファイルや、製品を構成する他のオブジェクト、たとえばディレクトリやアカウントは PDB に記録される。インストール時に選択した構成の設定も記録される。
製品テキスト・ファイル (PTF)	ソフトウェア・メーカーがオプションとして提供する POLYCENTER Software Installation ユーティリティ・ファイル。このファイルには、製品名、製品の開発者、構成の設定に関する説明、製品のインストールで使用されるメッセージ・テキストも含めて、製品に関する情報が格納される。
製品登録キー (PAK)	多数の弊社製品において、印刷物の形で通常提供される情報。この情報により、ソフトウェア・ ライセンス をシステムのライセンス・データベースに登録できる。
セーブ・セット	BACKUP ユーティリティで使用する特殊ファイル。BACKUP ユーティリティは、セーブ・セットにファイルをセーブしたり、セーブ・セットからファイルをリストアしたりする。インストールおよびアップグレード・プロセスでは、セーブ・セットからシステム・ディスクにプロダクト・ファイルをリストアする。
セクタ	Files-11 ディスク構造が認識できる最小単位。ほとんどの Files-11 ディスクの場合、セクタはブロックと同一である (512 バイト)。 ISO 9660 ボリューム上では、一意にアドレス指定できる単位で、CD-ROM 上の各セクタは、2,048 個の 8 ビット・バイトのシーケンスから構成される。
接続マネージャ	OpenVMS Cluster 環境において、OpenVMS Cluster システムを動的に定義し、各コンピュータのクラスタへの関与を調整するソフトウェア・コンポーネント。
設定モジュール 選択操作	プリント・ジョブにおいて、ファイルの先頭に挿入される デバイス制御モジュール 。 バージョン番号、ファイル・タイプ、UIC、作成日時、満了日付、変更日付などの基準に従って選択したファイルまたはボリュームを処理する BACKUP ユーティリティの動作。
選択ダンプ	クラッシュ・ダンプの分析に利用できるメモリ部分だけを記録する クラッシュ・ダンプ 。すべての物理メモリを記録するだけのディスク空間がない場合に有効である。 物理ダンプ を参照。
ターゲット・ディスク	VMSINSTAL.COM または VMSKITBLD.COM における、システム・ファイルの移動先ディスク。 ソース・ディスク を参照。
ターミナル・キュー	シンビオン を使用してターミナル・プリンタに出力を送る出力実行キュー。 プリンタ・キュー と サーバ・キュー を参照。
ターミナル・サーバ	ローカル・エリア・ネットワーク (LAN) と LAN に属する他のシステムに、ターミナル、モデム、プリンタを接続する通信デバイス。 LAT プロトコル を参照。
大容量記憶制御プロトコル・サーバ	OpenVMS Cluster 環境において、MSCP プロトコルを実現するコンポーネント。RA シリーズ・ディスクのような DSA ディスク用コントローラとの通信に使用される。ディスク・クラスの デバイス・ドライバ である DUDRIVER および DSDRVIER の一方または両方を組み合わせることで、MSCP サーバはコンピュータ上で MSCP プロトコルを実現し、そのコンピュータが記憶域制御デバイスとして機能することを可能にする。

チューニング	ある構成と作業負荷に対し、最適化された全体的性能を得るために各種のシステム値を変更する処理。
追加型バックアップ	BACKUP ユーティリティの動作。/RECORD 修飾子を使って行われた最新のバックアップの後で作成または変更されたファイルだけを保存する動作。/RECORD 修飾子により、ファイルがバックアップされた日付と時刻が記録される。
追加型リストアディスク	BACKUP ユーティリティの動作。追加型 セーブ・セット をリストアする。 ファイルが存在する物理的な媒体。
ディスク・クォータ	公用ボリューム上で各ユーザが利用できるディスク領域の制限を設定し、それを管理する方法。 クォータ・ファイル を参照。
データ領域	CD-ROM のボリューム空間を 2 つに分割したもののうちの 1 つで、論理セクタ 16 以降のボリューム空間が含まれる。
テープの終端マーカ	テープ・ボリュームの書き込み可能領域の終端を表す光反射性のマーク。
テープの先頭マーカ	テープ・ボリュームの書き込み可能領域の始まりを示す光反射性のマーク。
デバイス	記憶媒体へアクセスできるハードウェア。 ドライブ とも呼ばれる。
デバイス・ドライバ	特定の種類のデバイスの入出力動作を制御するシステム・コンポーネント。OpenVMS 上でデバイスを機能させるためには、そのデバイスを接続し、対応するデバイス・ドライバをメモリにロードする必要がある。
デバイス制御ライブラリ	ユーザが作成した、テキストまたはエスケープ・シーケンスから構成されるモジュールを格納するテキスト・ライブラリ。 デバイス制御モジュール も参照。
デバイス制御ライブラリ・モジュール	デバイス制御ライブラリ に含まれるユーザが作成したモジュールのこと。デバイス制御・ライブラリ・モジュールは次の目的に使用できる。 <ul style="list-style-type: none"> 指定されたポイント・サイズ、文字セット、書体 (ボールドまたはイタリック) などを設定するためのデバイスに依存したエスケープ・シーケンスを挿入する (プログラム可能プリンタで可能)。 プリント・ジョブの特定のポイントでテキストを挿入する (プログラム可能プリンタおよび非プログラム可能プリンタの両方で可能)。 設定モジュール、ページ設定モジュール、再設定モジュール も参照。
特権	ユーザがシステム上で実行できる機能を制限する手段。システム管理者の場合、一般のユーザには禁止されている特権が必要となる。
特権イメージ	既知イメージ 。このイメージを実行するプロセスには一時的に 特権 が追加され、イメージを実行する場合にかぎりユーザ登録ファイル (UAF) に定義されている特権制限を超えることができる。このため、通常の特権だけを持つユーザでも、通常の特権より高い特権を必要とするプログラムを実行できる。
ドライブ	記憶媒体へアクセスできるハードウェア。 デバイス とも呼ばれる。
トラック	ディスクの 1 記録面上で同一半径を持つ セクタ (Files-11 ボリュームではブロック) の集まり。ディスク・デバイスの特定の読み込みまたは書き込みヘッド位置でアクセス可能である。
トランザクション・グループ	特定の規則に従わなければならない SCSNODE 名を持ち、DECdtm トランザクションを行うコンピュータのグループ。
トレーラ・ラベル	ヘッダ・ラベル と似ているが、ファイルの後に書き込まれる磁気テープ・ラベル。
ネットワーク	複数のコンピュータを接続し、情報または通信をこれらのコンピュータが共有または転送することを可能とする手段。1 つのネットワークは、互いに接続された複数のコンピュータ、および、接続に使用するハードウェアとソフトウェアで構成される。
ネットワーク・プロキシ・アカウント	ネットワーク内の 遠隔ノード 上のユーザがローカル・システムのアカウントを利用してデータにアクセスできるようにするユーザ・アカウント。代理アカウントは、遠隔ノード上の 1 名以上のユーザに対し、特定のファイルに対するアクセスは許可するが、ローカル・システムのアカウントは禁止するという場合に有用である。
ネットワーク代理アカウント	ネットワーク・プロキシ・アカウント を参照。

ノンストップ・ブート	最も一般的なブート動作。システム・パラメータ値の変更などの特別な処理を終了しないままブートする場合に、ノンストップ・ブートを行う。 会話型ブート を参照。
パーティション	読み書き可能ディスクの論理サブセット。1つのディスクを複数のパーティションに分割し、そのそれぞれを独立して使用することができる。パーティションは、1つのディスクのように見える。
パーマネント・データベース	ネットワークにおける、 DECnet for OpenVMS 構成データベース の永久コピー。ネットワークを起動すると、このデータベースが 運用時データベース の初期値を提供する。パーマネント・データベースの変更は、ネットワークのシャットダウン後も有効であるが、現在のシステムには適用されない。
媒体	データを格納する物理的な実体。
バインド	InfoServer システムにおいて、ローカル OpenVMS システムに 仮想デバイス・ユニット を作成する機能。
バックリンク	Files-11 ディスク構造において、ファイルが存在するディレクトリを指すポインタのこと。
バッチ・ジョブ	1つまたは複数のコマンド・プロシージャを実行する独立プロセス。ユーザは、ジョブをバッチ・キューに登録する場合にコマンド・プロシージャのリストを定義する。
バッチ・モード	実行モードの1つ。このモードでは、コマンド・プロシージャをバッチ・キューに登録して実行することができる。必要な資源が使用可能な場合、システムは独立プロセスを作成してプロシージャ内のコマンドを実行する。通常、バッチ・モードで動作するプロセスは最も低いプロセス優先順位で実行される。これは、会話型ユーザとバッチ・ジョブがシステム資源を競合することを避けるためである。
バッチ実行キュー	バッチ・ジョブ専用の実行キューのこと。
バナー・ページ	各プリント・ジョブの先頭と最後、およびプリント・ジョブ内の各ファイルの先頭と最後に出力される特別に書式化されたページ。これらのページは、あるプリンタ上で出力された各ジョブを区別したり、1つのジョブ内で各ファイルを区別したりするときに利用される。
汎用キュー	バッチ・ジョブまたはプリント・ジョブを 実行キュー に転送するまでの間保持するキュー。各ジョブは、それを開始するための適切な実行キューが使用可能になるまで汎用キューに保持され、実行キューが使用可能になると、 キュー・マネージャ によってその実行キューに移される。
汎用出力キュー	汎用バッチ・キューの1つ。各ジョブをどの出力実行キューに送ることもできる。この種のキューは、特に出力のための作業負荷を複数の同機種のプリンタに分散するときに使用される。
汎用スタートアップ・コマンド・プロシージャ	システムがブートするたびに実行し、システムのスタートアップを管理するコマンド・プロシージャ。ファイル名は SYS\$STARTUP:STARTUP.COM であり、サイト別条件とはかかわりなく、すべてのシステムで必要である。このファイルを変更してはならない。 サイト別スタートアップ・コマンド・プロシージャ を参照。
汎用タイムシェアリング・サービス	LAN 上のユーザに処理資源を提供するサービス。 アプリケーション・サービス を参照。
汎用バッチ・キュー	汎用キューの1つ。各ジョブをバッチ用実行キューだけに送る。 汎用バッチ・キューは、特に OpenVMS Cluster 環境でバッチ・ジョブの作業負荷を複数のノードに分散するときに使用される。
必須アップデート	オペレーティング・システムのアップグレードおよびインストールの直後に必要なソフトウェアのアップデート。
非ローカル環境	SYSMAN ユーティリティにおいて、ローカル・ノード以外またはローカル・クラスタ以外で作業している場合の環境。
非ローカル・クラスタ	SYSMAN ユーティリティにおいて、SYSMAN を実行するクラスタ以外のクラスタ。
ファイル	Files-11 媒体上で、1～nの番号が付けられた、特定の属性を持つ連続した仮想ブロックの列。ファイルは、データ・ファイルまたはディレクトリ・ファイルのいずれかである。ディレクトリには、データ・ファイルとディレクトリ・ファイルの両方を含むことができる。

ファイル・バナー・ページ	1つのジョブ内で各ファイルを区別するための バナー・ページ 。各ユーザは、管理者がキューのために行ったバナー・ページの設定内容を変更できる。
ファイル・ヘッダ	Files-11 ボリューム上において、特定のファイルがボリューム上のどの部分を占有するかを示すもの。ファイル・ヘッダには、 所有者UID 、 保護コード 、作成日時、および ACL(アクセス制御リスト) などの情報が含まれる。
ファイル操作	BACKUP ユーティリティにおける、個々のファイルまたはディレクトリを処理する動作のこと。
フィードバック	システムが作業負荷を処理するために使用した各資源の量を示す情報で、 エグゼクティブ によって継続的に収集される。AUTGENをフィードバック・モードで実行すると、この情報が分析され、関連するすべてのシステム・パラメータの値が調整される。
フィールド	UAF レコードの AUTHORIZE ユーティリティによって変更を受ける部分。各フィールドに割り当てられた値に従って次のことが行われる。 <ul style="list-style-type: none"> • ユーザの識別 • ユーザの作業環境の定義 • システム資源の使用の制限
ブート	システム・ディスクからプロセッサ・メモリにシステム・ソフトウェアをロードするプロセス。ブートストラップとも呼ぶ。システムを初めてブートする場合には、OpenVMSオペレーティング・システムをインストールする必要がある。 会話型ブート および ノンストップ・ブート も参照。
ブート・ブロック	インデックス・ファイルの仮想ブロック 1。ブート (またはブートストラップ) ・ブロックはほとんどの場合、ボリュームの論理ブロック 0 にマッピングされる。ブート・ブロックには、 1次ブートストラップ・イメージ の場所とサイズが格納される。これは、システムをブートするときに使用される。特定のプロセッサはブートのためにブート・ブロックを読み込んで、1次ブート・ストラップ・イメージの場所を取得しなければならない。
ブートストラップ	ブート を参照。
物理セクタ	システムあるいはデータ領域の区分。ISO 9660 CD-ROM 上の最小のアドレス可能単位。
物理操作	BACKUP ユーティリティにおいて、ボリューム全体を論理ブロック単位でコピー、セーブ、リストア、比較する動作。ファイル構造は無視される。
物理ダンプ	物理メモリの内容全体を システム・ダンプ・ファイル に入れる クラッシュ・ダンプ 。 選択型ダンプ を参照。
プライベート・ボリューム	プライベート・ファイルだけが入っているファイル構造のディスク・ボリューム。
プリンタ・キュー	シンビオント によって出力をプリンタに送るタイプの出力実行キュー。 ターミナル・キュー と サーバ・キュー を参照。
プリント・ジョブ	プリンタに出力する1つ以上のファイルを指定する、出力キューのエントリ。出力するファイルは、プリント・ジョブを発行する場合にユーザが指定する。プリンタが使用可能状態になると、 キュー・マネージャ が シンビオント にファイルを送る。この結果、ファイルは書式化されて出力される。
プロセスの制限とクォータ	ユーザ・アカウントでプロセスが使用するシステム資源の量を制御するため、このアカウントに設定するユーザ登録ファイル (UAF) のパラメータ。UAFパラメータは、システム・パラメータとは異なる。AUTHORIZE ユーティリティによって値を設定する。
ブロック	Files-11 ディスク上において、ディスク空間の割り当てを行うときの基本単位 (1 バイトを 8 ビットとしたときの 512 バイト)。磁気テープではブロックのサイズはユーザによって指定される。
紛失ファイルページ	ロスト・ファイル を参照。 メモリの割り当ておよび解除で使用する単位。 VAX システムにおける 1 ページは 512 バイトである。 Alpha システムおよび I64 システムにおける 1 ページは 8 KB (8,192 バイト)、16 KB、32 KB、64 KB のいずれかである。Alpha コンピュータの初期段階のページ・サイズは、8,192 バイトである。 ページレット を参照。

ページ・ファイル	ページング動作において、メモリのページング部分をシステムが書き込むファイル。SYS\$SYSTEM:PAGEFILE.SYS と名付けられたページ・ファイルが、ディストリビューション・キットに入っている。システム・クラッシュ・ダンプ・ファイルの代わりに使用できる。
ページレット	Alpha システムおよび I64 システムにおける、512 バイトのメモリ単位。Alpha や I64 の 1 ページレットは、1 VAX ページと同じサイズである。また、Alpha 8KB コンピュータにおける 16 Alpha ページレットは 1 Alpha ページと同じである。
ページング	ディスクに入っているファイルと物理メモリとの間で情報を移動することにより、プロセスに割り当てた物理メモリを効率よく使用するメモリ管理動作。システムは、プロセス作業領域内の頻繁に使用されない部分を物理メモリからファイルに移動する。 スワッピング を参照。
ページ設定モジュール	プリント・ジョブの各ページの先頭に挿入される デバイス制御モジュール 。
ベクタ	VAX システムにおける、同じデータ・タイプであり、互いに関連性のある複数の スカラ値 、つまり要素で構成されるグループ。
ベクタ消費者	VAX システムにおいて、ベクタ機能を必要とし、ベクタ・コンテキストを持つプロセス。
ベクタ存在プロセッサ	VAX システムにおいて、VAX ベクタ処理システム構成に組み込まれた、スカラとベクタの統合化されたプロセッサ・ペア。
ベクタ対応システム	VAX システムにおいて、VAX ベクタ・アーキテクチャに準拠するシステム。
ヘッダ・ラベル	磁気テープ上で、ファイル名、作成日、満了日などの情報を含むラベル。磁気テープ上でファイルが作成されると、磁気テープ・ファイル・システムはすでに書き込み済みの最後のデータ・ブロックの直後にヘッダ・ラベルを書き込む。磁気テープ上の特定のファイルがファイル名によってアクセスされる場合、ファイル・システムは指定されたファイル名を含むヘッダ・ラベルをテープ上で検索する。
ヘッダ常駐イメージ	対応するイメージ・ファイルのヘッダがメモリ内に常駐する 既知イメージ 。ヘッダを常駐させることにより、一回のファイル・アクセスにおいてディスク入出力動作が 1 回だけ不要となる。
ホーム・ブロック	Files-11 ボリュームにおいてそれが Files-11 ボリュームであることを示すブロック。通常、 ブート・ブロック (ブロック 0) の次のブロックがホーム・ブロックとなる。何らかの理由でホーム・ブロックを読み込むことができない、すなわち物理的に利用できない場合は、ホーム・ブロックとして代替ブロックが使用される。代替ブロックには、ボリュームおよびボリューム上の各ファイルの省略時の値についての固有情報が格納される。
保護イメージ	共用可能イメージ であり、保護されたコードが入っている、 既知イメージ 。保護されたコードは、 カーネル または エグゼクティブ・モード で実行できるとともに、 ユーザ・モード・イメージ によって呼び出すことができるコードである。
保護コード	UIC に基づく保護 とともに使用されるコードであり、アクセスを誰にどのような理由で許可するかを示す。
保護論理名	エグゼクティブ・モード または カーネル・モード に対応する論理名。
保守リリース	アップデート・プロシージャ によって適用される OpenVMS オペレーティング・システムのリリース。
ホスト	TCP/IP ネットワークにおいて、TCP/IP に接続しているシステム。各ホストは、一意な名前とアドレスを持つ。ローカル・ホストは、ユーザがログインしているシステムである。遠隔ホストは、ユーザが通信を行っているシステムである。
ボリューム	新しいファイル構造を作成され、デバイス上にマウントされることによって使用可能状態となっているディスクまたはテープ。
ボリューム・セット	DCL の MOUNT/BIND コマンドによって 1 つの要素としてまとめられているディスク・ボリュームの集まり。ユーザには、1 つの大きなボリュームのように見える。 また、マルチボリューム・ファイルの集合が記録される複数のボリュームのこと。
ボリューム空間	ボリュームに関する情報を含む、ボリューム上のすべての論理セクタのセット。
マウント・チェック	ディスク操作およびテープ操作の回復メカニズム。マウント・チェックがオンの間、あるデバイスがオフラインになり、 ライト・ロック されても、その問題箇所を修正して、操作を続けることができる。

マスタ・エージェント	(eSNMP(Extensible Simple Network Management Protocol) を使用して) 管理対象のネットワーク上にあるルータやサーバなどのデバイス。マスタ・エージェントやサブエージェントは、 マネージャ と情報を交換する。マネージャは管理を行うネットワークにあるデバイスである。
マスタ・ファイル・ディレクトリ	ディスク上のすべてのユーザ・ファイル・ディレクトリの名前を保持するファイル。
マネージャ	eSNMP (Extensible Simple Network Management Protocol) を使用して管理を行うネットワーク上のデバイス。マネージャは マスタ・エージェント やサブエージェントと情報を交換する。これらのエージェントは管理対象のネットワーク上に存在するルータやサーバなどのデバイスである。
マルチボリューム・ファイル	1つのファイルのデータ・ブロックまたは関連性のある複数のファイルのデータ・ブロックが1つのボリュームに物理的に収まらないため、別のボリューム (磁気テープのリール) に渡るファイル。
満了日	Files-11 ディスク構造では、満了日によってファイルの使用が管理される。使用頻度の低いファイルを削除するときはそのファイルの満了日を利用できる。
密度	テープ上のデータの文字の測定値 (インチあたりのビット数)。
ユーザ環境テスト・パッケージ (UETP)	UETP を参照。
ユーザ識別コード (UIC)	ユーザ、ファイル、およびその他のシステム・オブジェクトに割り当てられている1組の数値で、使用できるアクセスのタイプ (所有者、グループ、一般ユーザ、システム) が指定される。UICは、グループ番号とメンバ番号をコンマで区切り、大括弧で囲んだ形式をとる。 アカウント と UICに基づく保護 も参照。
ユーザ登録ファイル (UAF)	システムへのアクセス権を許可した全ユーザのエントリを含むファイル。各エントリにより、システムを使用する個々のユーザに割り当てられている、ユーザ名前、パスワード、省略時のアカウント、UIC (ユーザ識別コード)、クォータ、制限値、および特権を指定される。
優先順位	基本プロセス優先順位 または ジョブ・スケジューリング優先順位 を参照。
ユーティリティ・プログラム	いくつかの関連操作を実行する弊社提供のプログラム。たとえば、バックアップ・ユーティリティ (BACKUP) では、ファイルの保存や回復を行うことができる。
ライセンス	多くのソフトウェア・ベンダはライセンスと呼ばれるもので自社のソフトウェアを顧客に提供する。ライセンスという用語は法律的な意味を持つことがあるが、このマニュアルでは製品の使用許可を指す。 ライセンス管理機能 (LMF) では、ソフトウェア・ライセンスの登録、管理、調査をオンラインで行うことができる。 製品登録キー (PAK) を参照。
リモート・ノード	ネットワークを通してローカル・ノードにアクセスできるノード。 SYSMAN ユーティリティでは、SYSMAN を実行しているノード以外のノード。 ローカル・ノード を参照。
隣接ノード	ネットワークにおいて、ローカル・ノードに一本の物理回線で接続されているノード。
ルーティング	複数のノードで構成されるネットワークにおいて、ソース・ノードからデスティネーション・ノード (エンド・ノード) にデータ・メッセージを送る処理。ルータとエンド・ノードは、ともにネットワーク内の他のノードとの間でメッセージの送受信を行える。 TCP/IP ネットワークにおいて、ルーティングとは、あるソース・ホストから他のホストへデータ・メッセージをダイレクトするプロセス。 ホスト 参照。
ループバック・テスト	ネットワークにおいて、そのネットワークが適正に動作するかどうかを調べるために行う一連のテスト。
レコード間ギャップ (IRG)	磁気テープの記録面上のデータ・レコード間に意識的に置かれたブランク。
レベル 1 ルータ	ネットワークにおいて、単一の領域内でルーティング動作を行うノード。 レベル 2 ルータ を参照。
レベル 2 ルータ	ネットワークにおいて、エリア間およびローカル・エリア内でルーティング動作を行うノード。 エリア・ルータ とも呼ばれる。 レベル 1 ルータ を参照。

ローカル・エリア VAXcluster 構成	VAXcluster 構成方法の 1 つ。管理センターとして機能する単一の VAX コンピュータと、このハブに接続されている 1 台以上のコンピュータとから構成される。
ローカル・クラスター	SYSMAN ユーティリティにおいて、SYSMAN を実行しているノードを指す。
ローカル・ノード	ネットワークにおいて、現在作業中のノードを指す。 SYSMAN ユーティリティでは、SYSMAN を実行するノードを指す。 遠隔ノード を参照。
ロード・アドレス	システムが ブート・イメージ をロードするメモリ内の位置。16 進表記で指定される。
ログイン・コマンド・プロシージャ	ユーザがログインするたびに実行されるコマンド・プロシージャ。このプロシージャには、ユーザ環境の設定などを行うコマンドを追加できる。
ログイン・システム・パラメータ	ログイン機能を制御するシステム・パラメータ。これらのパラメータの名前の先頭には LGI が付く。
ロスト・ファイル	ディレクトリにリンクされていないファイルのこと。ディレクトリ・ファイル (ファイル・タイプが .DIR のファイル) をその中のファイルを削除しないで削除すると、そのディレクトリによって参照されるファイルは紛失ファイルになる。紛失ファイルが存在するとディスク空間の無駄になり、ユーザの ディスク・クォータ に悪影響を与える。
論理キュー	汎用出力キューの特殊なもので、プリント・ジョブを他の出力実行キューに転送するもの。あるキューが稼働しているデバイスへの接続が切断された場合に、ジョブを一時的に別のキューに送るために使用する。
論理セクタ	ボリュームの編成単位。1 つまたは複数の物理セクタから構成される。1 つの物理セクタから開始できる論理セクタは 1 つだけである。 論理セクタは昇順で番号が付けられる。0 は、記録されたデータを含む最低位の物理アドレスを持つ論理セクタに割り当てられる。各論理セクタには、2,048 およびそれ以上のバイトからなるデータ・フィールドが含まれる (バイト数は常に 2 の累乗)。
論理ブロック	ボリューム空間の編成単位。論理ブロック・サイズは、論理セクタ・サイズを超えることはできない。
論理ブロックの番号付け	ボリューム空間の先頭バイトから始め、以降のボリューム空間に順次、昇順に番号を付けていくこと。
論理名テーブル	システム内のどのプロセスも使用できるシステム単位の論理名の定義を格納したテーブル。
論理リンク	ネットワークにおいて、2 つのプロセスを接続し、その 2 つのプロセス間を結ぶ サーキット 上で双方向トラフィックのストリームを搬送すること。2 つのノード間の単一のサーキットにより、多くの論理リンクを同時に確立できる。
割り当てクラス	OpenVMS Cluster 環境において、2 台のコンピュータ間でデュアル・ポート接続されたデバイスにはバスに依存しないデバイス名が割り当てられる。割り当てクラスは、そのようなデバイス名の一部として使用される。

索引

記号

- .LM\$JOURNAL ファイル・タイプ, 338
- 1 次スワップ・ファイル, 66
- 1 次プロセッサ, 328
- 1 次ページ・ファイル, 66
 - 記憶位置の必要条件, 107
- 2 次プロセッサ, 328

A

- ACCOUNTING コマンド, 220
- ACCOUNTING ユーティリティ, 220
- ACCOUNTING.DAT ファイル, 219
- ACL (アクセス制御リスト)
 - ベクタ機能オブジェクトでの設定, 333
- ADD_DUMPFILE シンボル, 93
- ADD_ERRORLOGDUMP シンボル, 93
- ADD_PAGEFILEn_SIZE シンボル, 94
- ADD_PAGEFILE シンボル, 93
- ADD_SWAPFILEn_SIZE シンボル, 94
- ADD_SWAPFILE シンボル, 93
- AGEN\$FEEDBACK_REQ_TIME 論理名, 43
- AGEN\$FEEDBACK.DAT ファイル
 - 説明, 32
- AGEN\$PARAMS.REPORT ファイル, 33
 - 管理, 33
 - 例, 33
- ALPHAVMSSYS.PAR ファイル, 25
 - ブート時のパラメータの初期化, 57
- ANALYZE/AUDIT コマンド, 181
 - (参照 監査分析ユーティリティ)
 - 機密保護レポートの作成, 201
- ANALYZE/DISK_STRUCTURE ユーティリティ
- BITMAP.SYS ファイルを作成する, 368
- VOLSET.SYS によってセットに属するボリュームを調べる, 369
 - クォータ・ファイルのコピーを作成する, 370
 - 使用するファイル, 365
 - ディレクトリ構造のファイルをチェックする, 369
 - ファイルの妥当性チェック, 368
- ATM を介した LAN エミュレーション
- デバイス・ドライバ, 256
- AUTHORIZE ユーティリティ
 - システムの使用制限, 103
 - プロセス・ページングの軽減, 333
 - ページ・ファイル使用量の制限, 71
 - ログインする時間帯の制限, 103
- AUTO_POSITIONING コマンド
- SHOW CLUSTER, 233
- AUTOGEN.COM コマンド・プロシージャ, 30
- ADD_ 接頭辞, 40
- AGEN\$PARAMS.REPORT ファイル, 33
- SETPARAMS.DAT ファイル, 39
- VAXcluster のノード数の定義, 42
- イーサネット・アダプタ数の指定, 42
- 影響されるシステム・パラメータ, 32

- 開始フェーズ
 - 起動時の指定, 31
- 起動, 31
- 機能, 30
- 計算結果の検討, 32
- 計算結果の調査, 39
- 計算されないパラメータの指定, 42
- システム・ダンプ・ファイルのサイズの計算, 63, 91, 92, 95
- システム・パラメータの変更, 25, 26
- システム・パラメータを変更するために推奨される方法としての, 27
- システム・ファイル, スワップ・ファイル, ダンプ・ファイルのサイズを指定
 - 個々のファイル空間, 92
 - ファイル空間の合計, 92
- 実行
 - 会話形式による, 39
 - バッチ・モードでの, 44
 - 実行される動作の制御, 31
- 実行モード
 - 起動時の指定, 31
- 収集されるデータの種類, 30
- 終了フェーズ
 - 起動時の指定, 31
- 使用するためのシステム・パラメータ値の変換, 27
- 使用法, 39
- スワップ・ファイルのサイズの計算, 66, 91, 92, 95
- 制限事項
 - ファイルのサイズの変更, 93
 - ページ・ファイル, スワップ・ファイル, ダンプ・ファイルのサイズの指定, 92
- 性能のチューニング, 105
- 設定したシステム・パラメータ値の制御, 40
- どんな場合に実行するか, 30
- パラメータ, 31
- 標準的な方法
 - ページ・ファイル, スワップ・ファイル, ダンプ・ファイルのサイズを変更する場合, 90
 - ファイルのサイズを計算するタイミング, 92
- フェーズ
 - 起動時の指定, 31
 - 実行順序, 37
- ページ・ファイル, スワップ・ファイル, ダンプ・ファイルのインストール, 91
- ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの記憶位置の指定, 93
- ページ・ファイル, スワップ・ファイル, ダンプ・ファイルのサイズ計算を表示, 91
- ページ・ファイル, スワップ・ファイル, ダンプ・ファイルのサイズの指定
 - 個々のファイル, 93
 - ファイル空間の合計, 93
- ページ・ファイル, スワップ・ファイル, ダンプ・ファイルのサイズの変更, 90

ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの作成, 93
ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの自動管理, 91
ページ・ファイルのサイズの計算, 64, 91, 92, 95
AUTOGEN の ADD_ 接頭辞, 40
AUTOGEN の MAX_ 接頭辞, 41
AUTOGEN の MIN_ 接頭辞, 41
AUTOGEN フィードバック, 30, 32, 44
影響を受ける資源, 32
格納されるファイル, 32
最小稼働時間, 33, 43
最大稼働時間, 33
システム作業負荷の重要性, 33
システム性能の向上, 32
収集, 32
チェック, 33
パラメータへの影響の調査, 33
レポート・ファイル, 33
自動送信, 44
自動的に送信, 30
例, 33
AUTOGEN ユーティリティ
予約メモリ・レジストリ, 125

B

BACKUP
システム・ダンプ・ファイルのコピー, 86
セーブ・セット・ファイルを MFD に入れる, 369
ダンプ・ファイルのコピー, 65
BACKUP.SYS ファイル, 363
(参照 バックアップ・ログ・ファイル)
Backup ユーティリティ (BACKUP)
システム・ダンプ・ファイルのコピー, 86
BADBLK.SYS ファイル, 363
(参照 不良ブロック・ファイル)
BADLOG.SYS ファイル, 363
(参照 保留不良ブロック・ログ・ファイル)
BAP システム・パラメータ, 57
BITMAP.SYS ファイル, 363
(参照 ストレージ・ビットマップ・ファイル)
BROADCAST デバイス設定, 195

C

CD-ROM
InfoServer Client for OpenVMS によるアクセス, 298
自動処理の形式, 289
CIPCA アダプタ
システム・チューニング, 57
CIXCD アダプタ
システム・チューニング, 57
CLUSTER_CONFIG.COM コマンド・プロシージャ, 224
LAN MOP の設定, 277
SATELLITE_PAGE.COM コマンド・プロシージャの作成, 89
CONFIGURATION SET コマンド
SYSMAN, 237
CONFIGURATION SHOW コマンド
SYSMAN, 237

CONTIN.SYS ファイル, 363
(参照 継続ファイル)
COPY コマンド
システム・ダンプ・アナライザ・ユーティリティ, 65, 85
ダンプ・ファイルをコピーするときの制限事項, 85
CORIMG.SYS ファイル, 363
(参照 コア・イメージ・ファイル)
CPU 識別番号, 328
CREATE コマンド
SYSGEN
ページ・ファイル, スワップ・ファイル, ダンプ・ファイルのサイズの変更, 96
ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの作成, 96
CTRLNAME 論理名, 174

D

DAD 仮想ディスク・ユニット, 302
DCL コマンド
SYSMAN の DO コマンドによる実行, 238
DCL 文
レポートからのチェック, 33
DECdns
完全な名前を格納, 250
ネーム・サービス
完全な名前を格納, 250
DECdim サービス
DECnet-Plus ネットワークでの使用, 340
開始, 358
停止, 357
トランザクション・ログの管理, 337
DECdim トランザクション・ログ
移動, 349
計画, 339
サイズのチェック, 346
サイズの変更, 347
作成, 342
定義, 338
DECnet
AUTOGEN の予測の上書き, 43
DECnet-Plus との比較, 250
DECnet over TCP/IP の使用, 245
許可, 252
UETP 中のエラー・メッセージ, 170
UETP テスト・フェーズ, 177, 178
UETP のための準備, 154
インストール時における UETP の省略時の設定, 172
説明, 244, 245
ノードの遠隔管理, 251
ノード名, 250
DECnet Phase IV (参照 DECnet)
DECnet Phase V (参照 DECnet-Plus)
DECnet-Plus
DECnet over TCP/IP の使用方法, 252
DECnet との違い, 244
DECnet との比較, 250
新しい機能, 250
移行

- 定義, 253
 - インストールおよび構成, 252
 - 開始と停止, 254
 - 完全な名前, 250
 - 管理ツール, 251
 - 関連ドキュメント, 254
 - クラスタのサポート, 251
 - 説明, 244
 - ネットワークに参加するための準備, 252
 - ネットワークの移行, 253
 - 利点, 250
 - 利点の要約, 249
 - DESELECT コマンド
 - SHOW CLUSTER, 234
 - Digital Network Architecture
 - Phase V (参照 DECnet-Plus)
 - DNA (Digital Network Architecture)
 - Phase V (参照 DECnet-Plus)
 - DOSD(ダンプ・オフ・システム・ディスク), 74, 76, 83
 - 必要条件, 81
 - DO コマンド
 - OpenVMS Cluster システムの管理用, 238
 - DUMP_DEV 変数, 74
 - DUMPPBUG システム・パラメータ, 64
 - DUMPPFILE_DEVICE シンボル, 92
 - DUMPPFILE シンボル, 93
 - DUMPPSTYLE システム・パラメータ, 64, 67, 72
- E**
- EDIT キーパッド機能, 229
 - ERLBUFFERPAGES システム・パラメータ, 67
 - ERRFMT (参照 エラー・フォーマット)
 - ERRFMT プロセス, 181
 - (参照 エラー・ログ・ファイル)
 - (参照 エラー・ログ・ユーティリティ)
 - (参照 エラー・ログ機構)
 - 再起動, 183
 - 削除されたときのメールの送信, 183
 - ERROR LOG, 181
 - (参照 ERRFMT プロセス)
 - (参照 エラー・ログ・ファイル)
 - (参照 エラー・ログ機構)
 - UETP との関係, 147, 160, 171
 - ERRORLOGBUFFERS システム・パラメータ, 67
 - ERRORLOGDUMP シンボル, 93
 - ESS\$LASTDRIVER デバイス・ドライバ, 297, 302
 - 制御と診断, 297, 298
 - ESS\$STARTUP.COM コマンド・プロシージャ, 298, 302
 - システム・スタートアップ時における起動, 298
 - Extended File Cache
 - 管理, 133
 - キャッシングの禁止, 137
 - サイズの制御, 134
 - 最大キャッシュ I/O サイズ, 137
 - 相互動作, 134
 - Extended File Cache (XFC)
 - OpenVMS Cluster での使用, 140
 - 先読みキャッシングの禁止, 138
 - 性能の監視, 138
 - 複合アーキテクチャ・クラスタでの使用, 140
- F**
- F\$GETJPI レキシカル関数
 - ベクタ処理に関する情報の入手, 334
 - F\$GETSYI レキシカル関数
 - ベクタ処理に関する情報の入手, 334
 - FDDI (Fiber Distributed Data Interface)
 - デバイス・ドライバ, 256
 - FDDI コントローラ
 - 複数コントローラでの問題, 300
 - FID (ファイル識別子), 363
 - Files-11
 - マスタ・ファイル・ディレクトリ (MFD), 365
 - 用語, 364
 - Files-11 オン・ディスク構造 (参照 Files-11 ディスク構造)
 - 構造
 - レベル 1, 365
 - レベル 2, 365
 - レベル 5, 365
 - Files-11 ディスク構造
 - ODS レベル 1, 2, 5 の比較, 370
 - UIC, 365
 - 構造, 364
 - セクタ, 364
 - ファイル識別子, 365
 - ブロック
 - 定義, 363
 - レベル 1
 - ディレクトリ階層および構造, 365
 - レベル 2
 - ディレクトリ階層および構造, 365
 - レベル 5
 - ディレクトリ階層および構造, 365
 - Files-11 メタデータ・キャッシュ, 131
- G**
- GBLPAGES システム・パラメータ, 120
 - GBLSECTIONS システム・パラメータ, 120
- H**
- HP TCP/IP Services for OpenVMS
 - インストールおよび構成, 248
 - 開始と停止, 248
 - 管理ツール, 247
 - 関連ドキュメント, 248
 - クラスタのサポート, 247
 - 説明, 244
 - ネットワークに参加するための準備, 247
 - 利点の要約, 246
- I**
- INDEXF.SYS ファイル, 363
 - (参照 索引ファイル)
 - InfoServer (参照 InfoServer Client for OpenVMS)
 - X ターミナル・クライアント, 292
 - X ターミナルのサポート, 292
 - 遠隔接続, 294

- 仮想デバイス・サーバ, 289
- 仮想デバイス・ユニット, 302
- 可用性, 292
- 管理セッションの終了, 295
- 機能, 289
- クライアント
 - および DECnet, 298
- クライアント・システムとの関係, 290
- クライアントの起動, 298
- コマンド, 295
- コンソール・ターミナル, 294
- サービスの切断, 292
- システム・スタートアップ時のセットアップ, 298
- システムの概要, 289
- 自動サービス, 292
- セッションの開始, 294
- ソフトウェア
 - Client for OpenVMS の起動, 298
 - Software Products Library からの更新, 291
 - クライアント・トランスポート (LAST) の問題, 300
 - クライアントの起動失敗, 299
- ダウンライン・ロード, 290
- デバイスのマウント
 - システム・スタートアップ時, 303
- 媒体の除去, 292
- フェールオーバー, 292
- 負荷分散, 294
- プロトコル, 293
- ヘルプ機能, 296
- マルチキャスト・アドレス機能, 293
- ローカル接続, 294
- InfoServer Client for OpenVMS
 - 機能, 296
 - コンポーネント, 296
 - ソフトウェア, 297
 - 自動起動, 298
 - スタートアップの制限事項, 301
 - 同一サービスへの高速アクセス, 292
- InfoServer Client for OpenVMS の起動, 298
- INSTALL (参照 イメージのインストール) (参照 既知イメージ)
 - システム性能の改善, 106, 116
 - 使用する理由, 116
 - ファイルへの同時アクセス数の表示, 106
- INSTALL コマンド
 - SYSGEN, 88
- Install ユーティリティ (INSTALL) (参照 イメージのインストール) (参照 既知イメージ)
 - アドレス・データ・セクションの内容の判断, 120
 - イメージの使用頻度の決定, 123
 - イメージの使用頻度の調査, 118
 - イメージを特権付きにする, 118, 121
 - イメージをヘッダ常駐にする, 117, 119
 - 永久オープン・イメージ, 117, 119
 - システム性能の改善, 106, 116, 117
 - 使用する理由, 116
 - ファイルへの同時アクセス数の表示, 106
- ISL, 290

K

- KFMSB アダプタ
 - システム・チューニング, 57

L

- LADCP ユーティリティ, 303
 - BIND コマンド, 303
 - 遠隔 InfoServer デバイスをローカルで使用可能にする, 303
 - 起動, 302
 - 終了, 302
 - ヘルプ機能, 303
 - 要約, 302
- LAD 制御プログラム, 289, 303
 - BIND コマンド, 303
 - 遠隔 InfoServer デバイスをローカルで使用可能にする, 303
 - 起動, 302
 - 終了, 302
 - ヘルプ機能, 303
 - 要約, 302
- LAD 制御プログラム (LADCP) ユーティリティ, 302
- LAN
 - CLUSTER_CONFIG による LAN MOP の設定, 277
 - DECnet MOP から LAN MOP への移行, 277
 - LANACP に関連する OPCOM メッセージ, 278
 - LANACP ユーティリティ, 259
 - LANCP SPAWN 機能, 262
 - LANCP コマンド・ファイル, 262
 - LANCP ユーティリティ, 260
 - LAN MOP 設定のサンプル, 278
 - LAN MOP と DECnet MOP, 276
 - LAN デバイス構成の表示, 263
 - LAN デバイス・パラメータの設定, 265
 - LAN デバイス・パラメータの表示, 263
 - MOP コンソール・キャリア, 282
 - MOP ダウンライン・ロード・サービス, 276, 279
 - MOP ダウンライン・ロード・サービスの許可, 279
 - MOP トリガ・ブート, 282
 - OPCOM メッセージの表示, 281
 - カウンタのクリア, 281
 - システム管理の強化, 255
 - 状態とカウンタの表示, 279, 280
 - デバイス管理, 262
 - デバイス情報の管理, 272
 - デバイス情報の削除, 273
 - デバイス情報の設定, 273
 - ノード情報の削除, 276
 - ノード情報の設定, 275
 - ノード情報の表示, 274
 - ノード・データベース管理, 274
 - ロード・トレース機能, 281
- LANACP ユーティリティ (参照 LAN 補助制御プログラム)
 - サーバ, 259
 - 実行, 260
 - 終了, 260
 - 表示される OPCOM メッセージ, 278
- LANCP (LAN 制御プログラム) ユーティリティ

- MOP ダウンライン・ロード・サービスの禁止, 279
- 状態とカウンタの表示, 279
- LANCP ユーティリティ, 258, 260
- LAN MOP の設定, 278
- MOP コンソール・キャリア, 282
- MOP ダウンライン・ロード・サービス管理, 279
- MOP ダウンライン・ロード・サービスの許可, 279
- MOP トリガ・ブート, 282
- OPCOM メッセージの表示, 281
- SPAWN 機能, 262
- カウンタのクリア, 281
- コマンド・ファイルの使用, 262
- 実行, 260
- 状態とカウンタの表示, 280
- デバイス管理, 262
- デバイス情報の削除, 273
- デバイス情報の設定, 273
- デバイス情報の表示, 272
- デバイス・データベース管理, 272
- ノード情報の削除, 276
- ノード情報の設定, 275
- ノード情報の表示, 274
- ノード・データベース管理, 274
- ロード・トレース機能, 281
- LAN 制御プログラム, 260
- LAN MOP の設定, 278
- MOP コンソール・キャリア, 282
- MOP ダウンライン・ロード・サービス管理, 279
- MOP ダウンライン・ロード・サービスの許可, 279
- MOP トリガ・ブート, 282
- OPCOM メッセージの表示, 281
- SPAWN 機能, 262
- カウンタのクリア, 281
- コマンド・ファイルの使用, 262
- 実行, 260
- 状態とカウンタの表示, 280
- デバイス管理, 262
- デバイス情報の削除, 273
- デバイス情報の設定, 273
- デバイス情報の表示, 272
- デバイス・データベース管理, 272
- ノード情報の削除, 276
- ノード情報の設定, 275
- ノード情報の表示, 274
- ノード・データベース管理, 274
- ロード・トレース機能, 281
- LAN ドライバ
- FDDI, 256
- アドレス, 257
 - ノード, 257
 - 物理, 257, 258
 - マルチキャスト, 258
- イーサネット, 256
- トークン・リング, 256
- プロトコルのタイプ, 256
- ポート, 256
- LAN フェイルオーバ, 283
- セット, 285, 287
- 特性, 287
- 例, 287
- LAN 補助制御プログラム
- サーバ, 259
- 実行, 260
- 終了, 260
- 表示される OPCOM メッセージ, 278
- LAN (ローカル・エリア・ネットワーク)
- LAT ノード用の複数の LAN アダプタ, 311
- LASTCP (参照 ローカル・エリア・システム・トランスポート制御プログラム)
- LASTport/Disk
- プロトコル, 293
- LASTport/Disk サービス, 302
- ESS\$DADDRIVER, 302
- LASTport/Tape
- プロトコル, 293
- LASTport/Tape サービス, 302
- ESS\$MADDRIVER, 302
- LASTport 制御プログラム (LASTCP)
- 機能, 297, 301
- LASTport 制御プログラム (LASTCP) ユーティリティ, 293, 302
- MAXBUF システム・パラメータの必要条件, 301
- アカウントの必要条件, 301
- 起動, 297
- コマンドの要約, 297
- 終了, 297
- トランスポート, 302
- 必要な特権, 297
- ヘルプ機能, 298
- LAT\$CONFIG.COM コマンド・プロシージャ, 318
- LAT\$STARTUP.COM コマンド・プロシージャ, 318, 319
- LAT\$SYSTARTUP.COM コマンド・プロシージャ, 318, 319, 320, 324
- 例, 324
- LATACP (LAT 補助制御プロセス), 326
- LATCP ユーティリティ, 305, 307, 316
- ANNOUNCEMENTS 修飾子, 308
- LARGE_BUFFER 修飾子, 316
- SET SERVICE コマンド, 323
- 外部からの接続要求のキュー登録, 323
- 起動, 317
- コマンドのまとめ, 317
- 終了, 317
- 制限されたサービスの設定, 322
- 特徴, 317
- LAT サーバ, 294
- LAT 制御プログラム, 305, 307, 316
- ANNOUNCEMENTS 修飾子, 308
- LARGE_BUFFER 修飾子, 316
- SET SERVICE コマンド, 323
- 外部からの接続要求のキュー登録, 323
- 起動, 317
- コマンドのまとめ, 317
- 終了, 317
- 制限されたサービスの設定, 322
- 特徴, 317

- LAT ソフトウェア, 305
 - LAT\$STARTUP.COM によるスタートアップ, 318, 319
 - アプリケーション・プログラム, 306
 - 外部からの接続要求のキュー登録, 323
 - 外部への接続, 305, 307, 316
 - 外部への接続の許可, 324
 - カスタマイズ, 320
 - コマンド・プロシージャによるネットワークの起動, 319
 - サービス
 - 遠隔プリント, 305
 - 専用のアプリケーション, 305
 - 通知, 308, 316
 - 定義, 305
 - データベース, 316
 - ノード, 307, 317
 - サービス通知の無効化, 308
 - サービスの定義, 321
 - 制限されたサービスの設定, 322
 - ターミナル, 306
 - 大容量のバッファの使用, 315
 - データベースのサイズの管理, 326
 - 負荷分散, 306
 - 複数の LAN アダプタの使用, 311
 - プリンタ, 306
 - モデム, 306
 - 利点と用途, 306
 - 論理ポートの設定, 321
 - LAT ソフトウェアのスタートアップ
 - LAT\$STARTUP.COM による, 318, 319
 - LAT データベース・サイズの管理, 326
 - LIBDECOMP.COM コマンド・プロシージャ, 106, 108
 - 影響を受けるライブラリ, 109
 - 会話型での使用, 110
 - バッチ・モードでの使用, 115
 - パラメータ, 110
 - LIBRARY コマンド
 - ライブラリの展開または縮小に使用, 115
 - LOADS 論理名, 177
 - LOGOUT コマンド, 335
 - LTDRIVER
 - オンおよびオフ, 317
- M
- MAD 仮想テープ・ユニット, 302
 - MAX_DUMPFILe シンボル, 93
 - MAX_ERRORLOGDUMP シンボル, 93
 - MAX_PAGEFILEn_SIZE シンボル, 94
 - MAX_PAGEFILE シンボル, 93
 - MAX_SWAPFILEn_SIZE シンボル, 94
 - MAX_SWAPFILE シンボル, 93
 - MFD, 363, 365
 - BACKUP がセーブ・セット・ファイルを入れる, 369
 - 説明, 369
 - 予約ファイル, 369
 - MIN_DUMPFILe シンボル, 93
 - MIN_ERRORLOGDUMP シンボル, 93
 - MIN_PAGEFILEn_SIZE シンボル, 94
 - MIN_PAGEFILE シンボル, 93
 - MIN_SWAPFILEn_SIZE シンボル, 94
 - MIN_SWAPFILE シンボル, 93
 - MODE 論理名, 161, 180
 - MODPARAMS.DAT ファイル, 38, 40
 - ADD_ 接頭辞, 40
 - AUTOGEN が設定したパラメータ値の制御, 27, 39
 - MAX_ 接頭辞, 41
 - MIN_ 接頭辞, 41
 - 外部パラメータ・ファイルの取り込み, 43
 - システム・パラメータの変更内容の保存, 27
 - パラメータ値の指定
 - 最小値, 41
 - 最大値, 41
 - 絶対値, 42
 - パラメータの値を大きくする方法, 40
 - ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの作成, 93
 - ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの制御, 91, 92
 - 記憶位置の指定, 93
 - 個々のファイルのサイズの指定, 93
 - ファイル空間の合計サイズを指定, 93
 - 例, 39
 - ログ取得文, 33
 - MONITOR.COM コマンド・プロシージャ, 211
 - Monitor ユーティリティでの使用, 211
 - MONITOR コマンド
 - 出力先の指定, 207
 - スクリーン・イメージの表示間隔の指定, 208
 - 動作時間の記録, 208
 - 入力ファイルの指定, 208
 - ノードの指定, 207
 - 表示時間の指定, 208
 - 表示内容の保存, 209
 - ファイル・システムおよびプロセスに関するデータの記録, 208
 - ログ要求の実行方法の指定, 206
 - Monitor ユーティリティ
 - MONITOR.COM コマンド・プロシージャ
 - 要約ファイルの作成, 211
 - MONSUM.COM コマンド・プロシージャ
 - クラスタ全体の要約レポートの作成, 211
 - SUBMON.COM プロシージャ
 - 独立プロセスとしての MONITOR.COM の起動, 211
 - 起動, 205
 - 記録した動作の遠隔ブレイバック, 210
 - 記録した動作のブレイバック, 208
 - 記録ファイルの更新, 210
 - クラスの種類, 203
 - 継続実行, 211
 - コマンドの入力, 205
 - コマンド・プロシージャ, 211
 - クラスタの要約の作成, 211, 213
 - 継続記録の開始, 213
 - システム・ディスクの入出力を軽減するためのログ・ファイルの移動, 107
 - システムの動作の記録, 207
 - システムの動作の表示, 206

システムの動作の表示と記録, 208
 修飾子, 205
 終了, 205
 説明, 203
 バージョンの互換性, 215
 パラメータ, 203
 表示先の指定, 205
 レポートの生成, 211
 MONITOR ユーティリティ
 レポート, 211
 Monitor ユーティリティ (MONITOR), 101
 (参照 MONITOR コマンド)
 MONSUM.COM コマンド・プロシージャ, 213
 Monitor ユーティリティでの使用, 211
 クラスタ全体の要約レポートの作成, 211
 MOP (保守操作プロトコル)
 CLUSTER_CONFIG.COM による設定, 277
 DECnet MOP から LAN MOP への移行, 277
 DECnet MOP との共存, 276
 LAN MOP, 276
 LANACP サーバ, 279
 LAN MOP 設定のサンプル, 278
 カウンタのクリア, 281
 機能, 252
 許可, 279
 禁止, 279
 コンソール・キャリア, 282
 状態とカウンタの表示, 279, 280
 ダウンライン・ローディング, 290
 トリガ・ブート, 282
 MOUNT コマンド
 SYSMAN の DO コマンドと /CLUSTER をともに使用
 しないようにする, 239
 システム・スタートアップ時
 遠隔 InfoServer デバイス, 303
 ページ・ファイルとスワップ・ファイルを保持する
 ディスクのマウント, 89
 MOVE キーパッド機能, 229
 MSCP, 224
 MSCP サーバ, 224
 MULTIPROCESSING システム・パラメータ, 329

 N
 NCL, 250
 NCP
 管理
 Phase IV ノードの, 251
 NET\$CONFIGURE
 DECnet-Plus の構成, 252
 NET\$PROXY.DAT ファイル
 システム・ディスクの入出力を軽減するための移動,
 107
 NET\$PROXY 論理名
 システム・ディスクの入出力を軽減するための定義,
 107
 NETPROXY.DAT ファイル
 システム・ディスクの入出力を軽減するための移動,
 107
 NETPROXY 論理名

 システム・ディスクの入出力を軽減するための定義,
 107
 Network Control Language (参照 NCL)
 NPAGEDYN システム・パラメータ, 333
 NUM_ETHERADAPT シンボル, 42
 NUM_NODES シンボル, 42

 O
 OLDUETP.LOG ファイル, 164
 OPC\$LOGFILE_CLASSES 論理名, 194, 195
 OPC\$LOGFILE_ENABLE 論理名, 195
 OPC\$LOGFILE_NAME 論理名, 195
 オペレータ・ログ・ファイルを表す, 193
 OPC\$OPAO_ENABLE 論理名, 195
 OPCOM, 181
 (参照 オペレータ・ログ・ファイル)
 オペレータ・ログ・ファイルの再起動, 196
 オペレータ・ログ・ファイルの省略時の状態の指定,
 195
 オペレータ・ログ・ファイルの設定, 193
 オペレータ・ログ・ファイルのプリント, 196
 クラス
 ログ・ファイルに記録するクラスの使用可能化と使
 用禁止, 194
 ログ・ファイル, 188
 OPCOM メッセージ
 LANACP 起動時に表示される, 278
 LANACP による生成, 281
 SYSGEN, 191
 オペレータの応答, 190
 機密保護アラーム, 192
 ユーザ要求, 190
 OpenVMS Cluster 環境
 CLUSTER_CONFIG.COM の使用法, 224
 OpenVMS Cluster 環境内の機密保護監査ログ・ファ
 イル, 202
 SHOW CLUSTER による監視, 226
 SYSMAN による管理, 237
 UETP 中のテスト障害, 168
 UETP のための準備, 153
 機密保護, 237
 共有資源, 224
 システム管理, 225
 デュアル・アーキテクチャ
 イメージのインストール, 239
 プリント・キューとバッチ・キュー, 224
 利点, 224
 OpenVMS Cluster システム
 DECnet MOP から LAN MOP への移行, 277
 DECnet ノードの構成, 251
 LAN MOP 設定のための CLUSTER_CONFIG の使用,
 277
 LAN MOP の DECnet MOP との共存, 276
 LAN MOP 用の設定, 278
 LAN 管理の強化, 255
 OpenVMS Cluster システム内の機密保護監査ログ・
 ファイル, 202
 TCP/IP のサポート, 247
 共通なパラメータ・ファイル, 43

クロス・アーキテクチャ・ブート, 279
問題点と制限事項
BAP システム・パラメータのチューニング, 57
OPERATOR.LOG ファイル (参照 オペレータ・ログ・ファイル)

P

P1 空間

IMGREG_PAGES システム・パラメータがサイズを決定, 120
共用アドレス・データの割り当て, 120
PAGEFILE.SYS ファイル, 63, 65
システム・ダンプ・ファイルとして, 67
必要条件
記憶位置, 107
要求される記憶位置, 64
PAGEFILEn_NAME シンボル, 92, 94
PAGEFILEn_SIZE シンボル, 94
PAN キーパッド機能, 229
PAN コマンド, 228
PARAMETERS コマンド
SYSMAN, 47
要約, 51
PATHWORKS
スタートアップの制限事項, 301
PCF (製品構成ファイル), 393
PDF (製品記述ファイル), 393
PGFLQUOTA プロセス制限
標準的な最小値, 71
ページ・ファイルの使用量の制限, 71
PTF (製品テキスト・ファイル), 393

Q

Qlogic 1020ISP アダプタ
システム・チューニング, 57
QMAN\$MASTER 論理名
システム・ディスクの入出力を軽減するための定義, 107
QUANTUM システム・パラメータ, 333
QUOTA.SYS ファイル (参照 クォータ・ファイル)

R

REPLY/ENABLE=SECURITY コマンド
セキュリティ・オペレータ・ターミナルを使用可能にする, 200
REPLY コマンド
新しいオペレータ・ログ・ファイルのオープン, 188, 194
オペレータ・ターミナルの使用開始, 189
ターミナルの指定, 189
オペレータ・ターミナルの使用停止, 189
オペレータ・ログ・ファイルに記録されない応答, 190
現在のオペレータ・ログ・ファイルのクローズ, 188
ユーザ要求の取り消し, 190
REQUEST コマンド
オペレータへの要求の送信, 190
オペレータ・ログ・ファイルへの要求の記録, 190
RIGHTSLIST.DAT ファイル

システム・ディスクの入出力を軽減するための移動, 107
RIGHTSLIST 論理名
システム・ディスクの入出力を軽減するための定義, 107
RMS_EXTEND_SIZE システム・パラメータ, 106
RSM (リモート・システム・マネージャ)
スタートアップの制限事項, 301
RUN コマンド
インストール済みイメージの影響, 117
RV60 光ディスク・ドライブ
UETP によるサポート, 151
RVN (相対ボリューム番号), 363
(参照 ファイル識別子)

S

SATELLITE_PAGE.COM コマンド・プロシージャ, 89
SAVEDUMP システム・パラメータ, 63, 64, 67
SCB, 363
SCROLL キーパッド機能, 229
SCROLL コマンド, 235
SCSNODE システム・パラメータ, 298
DECnet-Plus ネットワーク, 340
SDA CLUE
ダンプ・オフ・システム・ディスク, 83
SDA CLUE コマンド
システム・ダンプ・ファイルの内容の保存, 85
システム・ダンプ・ファイルの分析, 62
ダンプ・ファイル情報の収集, 82
ダンプ・ファイル情報の保管, 82
SECURITY_AUDIT.AUDIT\$JOURNAL ファイル
システム・ディスクの入出力を軽減するための移動, 107
SECURITY.AUDIT\$JOURNAL ファイル (参照 機密保護監査ログ・ファイル)
SELECT コマンド
SHOW CLUSTER, 234
SEQ, 363
(参照 ファイル識別子)
SET ACCOUNTING コマンド, 219
会計情報ファイルの更新, 219
調査対象の資源の制御, 219
SET AUDIT コマンド
機密保護アラームを使用可能にする, 200
機密保護監査機構を使用可能にする, 197
機密保護監査ログ・ファイルの新しいバージョンの作成, 201, 202
ローカル・ノード上のみ, 202
SET FUNCTION コマンド, 229
SHOW CLUSTER, 234
SET HOST/LAT コマンド, 305
SET LOGINS コマンド, 103
SETPARAMS.DAT ファイル, 39
SET QUORUM コマンド
SYSMAN の DO コマンドと /CLUSTER をともに使用しないようにする, 239
SET SECURITY/ACL コマンド
ベクタ機能 (VAX), 334
SET VOLUME コマンド

ハイウォータ・マークの無効化, 106
 SET (フィールド) コマンド
 SHOW CLUSTER, 233
 SHOW_CLUSTER\$INIT コマンド・プロシージャ, 235
 SHOW_CLUSTER\$INIT 論理名, 236
 SHOW ACCOUNTING コマンド, 218
 SHOW AUDIT コマンド
 監査されているイベント・クラスを表示する, 199
 SHOW CLUSTER ユーティリティ
 画面のリフレッシュ, 232
 キーボードのキーの定義, 229
 使用法, 226
 スタートアップ初期化ファイル, 228, 235
 表示内容の制御, 228, 232
 レポート, 227
 初期化ファイル, 231
 書式化, 228, 232, 235
 表示内容の制御, 231
 SHOW CPU コマンド, 330, 335
 SHOW DEVICES コマンド
 ファイルの状態の調査, 124
 SHOW MEMORY コマンド
 非ページング・プールのサイズの表示, 333
 ページ・ファイル, スワップ・ファイル, ダンプ・
 ファイルのサイズ, 93
 ページ・ファイルとスワップ・ファイルの表示, 66
 ページ・ファイルの使用状況の監視, 66
 SHOW PROCESS コマンド, 335
 SHOW SECURITY コマンド, 334
 SHOW コマンド
 SYSGEN, 52
 SMP_CPUS システム・パラメータ, 329, 331
 Software Products Library
 InfoServer ソフトウェアに更新, 291
 SPAWN 機能
 LANCP (LAN 制御プログラム) ユーティリティ, 262
 START/CPU コマンド, 329, 332
 STARTUP.COM コマンド・プロシージャ
 行うタスク, 117
 STOP/CPU コマンド, 329, 332
 SUBMON.COM コマンド・プロシージャ
 Monitor ユーティリティ での使用, 211
 例, 213
 SWAPFILE.SYS ファイル, 65
 SWAPFILEn_NAME シンボル, 92, 94
 SWAPFILEn_SIZE シンボル, 94
 SWAPFILES.COM コマンド・プロシージャ
 1 次ページ・ファイル, 1 次スワップ・ファイル, 1
 次ダンプ・ファイルのサイズの変更, 94
 SWAPFILE シンボル, 93
 SYLOGICALS.COM コマンド・プロシージャ
 オペレータ・ログ・ファイルの省略時の状態の指定,
 195
 システム・ファイルの記憶位置の再定義, 107
 SYPAGSWPFILES.COM コマンド・プロシージャ, 88,
 89
 SYS\$AUDIT_SERVER_INHIBIT 論理名, 199
 SYS\$DECDTM_INHIBIT 論理名, 358
 SYS\$ERRLOG.DMP ファイル, 63
 SYS\$ERRORLOG 論理名
 システム・ディスクの入出力を軽減するための定義,
 107
 SYS\$INPUT 論理名, 174
 SYS\$JOURNAL 論理名, 338
 SYS\$MONITOR 論理名
 システム・ディスクの入出力を軽減するための定義,
 107
 SYS\$OUTPUT 論理名, 175
 SYS\$TEST 論理名, 148, 154, 164
 SYSDUMP.DMP ファイル, 63
 保護, 65
 要求される記憶位置, 64
 SYSMAN ユーティリティ
 予約メモリ・レジストリ, 125
 SYSTARTUP_VMS.COM コマンド・プロシージャ
 InfoServer Client for OpenVMS の起動, 298
 遠隔 InfoServer デバイスを使用可能にする, 303
 既知イメージのインストール, 117
 システム・ダンプ・ファイルの内容の保存, 85
 システム・ダンプ分析ユーティリティの呼び出し, 85
 ダンプ情報のページ・ファイルの解放, 88
 ページ・ファイルからのダンプ情報の解放, 65
 System Dump Analyzer ユーティリティ (SDA)
 リポート時に自動起動, 82
 SYSTEST_CLIG アカウント
 UETP のための再使用可能, 153
 UETP のための要件, 153, 179
 SYSTEST アカウント
 UETP のためのログイン, 146, 147
 UETP を実行するのに必要なクォータ, 166
 UETP を実行するのに必要な特権, 166
 SYSTEST ディレクトリ
 UETP 中の機能, 148
 UETP のための作成, 150
 SYSUAF.DAT ファイル
 システム・ディスクの入出力を軽減するための移動,
 107
 SYSUAF 論理名
 システム・ディスクの入出力を軽減するための定義,
 107

T
 TCP/IP (Transmission Control Protocol/Internet Protocol),
 247
 TCPIP\$CONFIG
 HP TCP/IP Services for OpenVMS の構成, 248
 TLZ04 テープ・ドライブ
 UETP によるサポート, 151
 TMSCP, 224
 TP_SERVER, 337
 TP_SERVER プロセス
 永久停止, 358
 ディスクのディスマウントの終了, 352

U
 UETCONT00.DAT ファイル, 173
 作成, 174
 UETINIDEV.DAT ファイル, 173, 174

- 形式, 174, 175
- 作成, 174
- UETININET.DAT ファイル, 178
- UETINIT00.EXE イメージ, 173
- UETINIT01.EXE イメージ, 164, 173
- UETLOAD00.DAT ファイル, 176
- UETNETS00.EXE ファイル, 178
- UETP.COM プロシージャ, 173
 - 終了, 159
- UETP.LOG ファイル, 158, 164, 171, 177
- UETP\$NODE_ADDRESS 論理名, 154
- UETPHAS00.EXE プログラム, 173
- UETP 中のエラー, 164
 - 原因, 161
 - 診断, 160
- UETP 中のコンソール・レポート, 158
- UETP 中の出力
 - ターミナルおよびライン・プリンタ, 175
- UETP(ユーザ環境テスト・パッケージ)
 - DECnet インストール時の省略時の設定, 172
 - TLZ04 テープ・ドライブ, 151
 - VAX ベクタ命令, 156
 - 起動, 157
 - 個々のフェーズの実行, 157
 - コンパクト・ディスク・ドライブのサポート, 151
 - システム資源の要件, 146, 148
 - 実行時のテストの表示, 161
 - 実行の中断, 159
 - 出力の中断, 160
 - 小規模ディスク・システムのための要件, 154
 - 初期化フェーズ, 173
 - すべてのフェーズの実行, 146
 - 正常終了, 159
 - 説明, 145
 - 光ディスク・ドライブのサポート, 151
 - 必要なクォータ, 166
 - 必要な特権, 166
 - 複数パスの実行, 157, 164
 - ベクタ・プロセッサのテスト, 156
 - 編成, 173
 - 報告される一般的な障害, 164
 - マスタ・コマンド・プロシージャ, 173
- UETP 用フェーズ・コントローラ (参照 UETPHAS00.EXE プログラム)
- UETUNAS00.EXE UETP テスト・イメージ, 163
- UFD, 363
- UFD (ユーザ・ファイル・ディレクトリ), 365
 - MFD に含められた, 365
 - 対応する UIC, 365
- UIC
 - UETP のための, 150
- UIC (利用者識別コード)
 - クォータ・ファイルに保存されるディスク使用量, 369
- USE コマンド
 - SYSGEN, 52, 54

V

- VAXcluster 環境
 - AUTOGEN のノード数の定義, 42

- VAXVMSSYS.PAR ファイル, 25
 - ブート時のパラメータの初期化, 57
- VAX ベクタ命令エミュレーション機能
 - アンロード, 335
 - 存在するかどうかの判断, 335
 - ロード, 335
- VAX ベクタ命令エミュレーション機能 (VVIEF), 156
- VCN_CACHE_S システム・パラメータ, 132, 143
- VCC_FLAGS システム・パラメータ, 132, 143
- VCC_MAX_CACHE システム・パラメータ, 135
- VCC_MAX_IO_SIZE システム・パラメータ, 137
- VCC_MAXSIZE システム・パラメータ, 141, 144
- VCC_PTES システム・パラメータ, 142
- VCC_READAHEAD システム・パラメータ, 138
- VCR (ベクタ・カウンタ・レジスタ), 330
- VECTOR_MARGIN システム・パラメータ, 333
- VECTOR_PROC システム・パラメータ, 331
- Virtual I/O Cache, 140, 144
 - OpenVMS Cluster の構成, 144
 - VIOC の許可, 143
 - VIOC のサイズの調整, 144
 - VIOC の選択, 141
 - 管理, 140
 - キャッシュのサイズの管理, 141
 - キャッシング・ポリシー, 141
 - 許可, 143
 - 禁止, 143
 - サイズの調整, 144
 - 統計情報の表示, 142
 - パラメータ
 - VCN_CACHE_S, 143
 - VCC_FLAGS, 143
 - VCC_MAXSIZE, 144
 - メモリ再生, 144
 - メモリの割り当て, 144
 - 目的, 141
- VLR (ベクタ長レジスタ), 330
- VMR (ベクタ・マスク・レジスタ), 330
- VMScluster システム
 - 遠隔監視の制限, 215
- VMSMAIL_PROFILE.DATA ファイル
 - システム・ディスクの入出力を軽減するための移動, 107
- VMSMAIL_PROFILE 論理名
 - システム・ディスクの入出力を軽減するための定義, 107
- VOLSET.SYS ファイル, 363
 - (参照 ボリューム・セット・リスト・ファイル)
- VVIEF (参照 VAX ベクタ命令エミュレーション機能)
 - アンロード, 335
 - 存在するかどうかの判断, 335
 - ロード, 335
- VVIEF\$DEINSTALL.COM コマンド・プロシージャ, 335
- VVIEF\$INSTALL.COM コマンド・プロシージャ, 331, 335
- VVIEF (VAX ベクタ命令エミュレーション機能)
 - 定義, 331

W

- WSMAX システム・パラメータ, 333

X

XA Gateway, 359

XFC, 133

OpenVMS Cluster での使用, 140
管理, 133

キャッシングの禁止, 137

サイズの制御, 134

最大キャッシュ I/O サイズ, 137

先読みキャッシングの禁止, 138

性能の監視, 138

相互動作, 134

複合アーキテクチャ・クラスタでの使用, 140

X ターミナル・クライアント, 292

あ

アーカイブ・ファイル

クラスタ内の各ノード用に作成, 198

アクセス制御リスト

ベクタ機能オブジェクトでの設定, 333

アクティブ・システム・パラメータ, 25, 47

アクティブ・セット, 328

表示, 330

アベイラブル・セット, 328

アラーム

機密保護

使用可能にする, 200

アラーム・メッセージ

機密保護, 192

い

イーサネット

UETP 用の遠隔ノードの定義, 163

アダプタ

MODPARAMS.DAT ファイルでの数の指定, 42

コントローラ

複数コントローラでの問題, 300

デバイス・ドライバ, 256

イベント・クラス

監査中の表示, 199

イメージ, 116

(参照 既知イメージ)

アプリケーション, 120

インストール

RUN コマンドへの影響, 117

アプリケーション, 120

共用アドレス・データ付きの, 120

共用アドレス・データによる, 119

システム・スタートアップ時, 117

システムの性能の改善, 106

理由, 116

インストールされたイメージの保護, 122

永久オープン, 117, 119

書き込み可能, 118

既知, 117

共用可能, 118, 120, 121

論理名の割り当て, 124

システム提供の, 120

実行可能, 117, 121, 122

実行専用, 122

常駐 (Alpha), 118

使用頻度の決定, 123

使用頻度の調査, 118

定義, 116

特権, 118, 121

機密保護についての注意, 121

特権付き共用可能, 121

複数ユーザによる同時アクセス, 118

ヘッダ常駐, 117, 119

保護, 118, 121

保護モードでの動作, 118, 121

ユーザ・レベル

保護されたコードの呼び出し, 118, 121

リンク可能, 117

イメージのインストール, 122

RUN コマンドへの影響, 117

SYSTARTUP_VMS.COM コマンド・プロシージャ, 117

システムの性能の改善, 106, 117

理由, 116

インストールとアップグレード情報

Alpha のみ

問題点と制限事項, 57

インストール・ファイル (参照 既知イメージ)

う

上書き

DECnet パラメータ, 43

え

エクステン

索引ファイル

説明, 368

定義, 363

エグゼクティブ特権モード

動作中のイメージの呼び出し, 121

エグゼクティブ・モード

動作中のイメージの呼び出し, 118

論理名, 122

エラー

エラー・レポートの分析, 182

エラー・ログ機構, 182

エラー・ログ・ファイル, 182

エラー・チェック

システム・パラメータ・ファイル内, 40

エラー・フォーマット

説明, 182

メールの使用可能, 184

メールの使用不能, 184

メールの送信, 183

メール・メッセージによるユーザへの通知, 184

メール・ユーザ名の変更, 184

エラー・ログ

機構, 182

生成されるレポート, 182

エラー・ログ機構, 181

(参照 ERRFMT プロセス)

(参照 エラー・ログ・ファイル)

(参照 エラー・ログ・ユーティリティ)

説明, 182

エラー・ログ・ダンプ・ファイル, 62
AUTOGEN が作成する, 63
インストール
自動的に, 63
サイズ
AUTOGEN による計算, 63
AUTOGEN による操作, 92
机上での計算, 68
複数のファイルの合計を指定, 93
作成
AUTOGEN による, 93
作成後のリポート, 97
定義, 62
エラー・ログ・ファイル
管理, 183
システム・ディスクの入出力を軽減するための移動,
107
エラー・ログ・ユーティリティ, 181
(参照 ERRFMT プロセス)
(参照 エラー・ログ・ファイル)
(参照 エラー・ログ機構)
UETP との関係, 147, 160, 171
エラー・ログ・レポート (参照 エラー・ログ・ユーティ
リティ) (参照 レポート)
遠隔 InfoServer デバイス
BIND コマンド, 303
システム・スタートアップ時のマウント, 303
使用可能にする, 303
遠隔監視
混在バージョンの VMScLuster システム, 215
制限, 215
遠隔ログ・(アーカイブ)・ファイル, 198

お
オープン・イメージ, 117, 119
オペレータ
機密保護ターミナル, 192
オペレータ・クラス, 181, 194 (参照 OPCOM)
オペレータ (コンピュータ)
機密保護ターミナル, 192
オペレータ・ターミナル
使用開始と使用停止, 189
使用可能化と使用禁止
機密保護アラーム, 200
オペレータ通信マネージャ, 181
(参照 OPCOM メッセージ)
オペレータ・ログ・ファイルの再起動, 196
オペレータ・ログ・ファイルの省略時の状態の指定,
195
オペレータ・ログ・ファイルの設定, 193
オペレータ・ログ・ファイルのプリント, 196
クラス
ログ・ファイルに記録するクラスの使用可能化と使
用禁止, 194
ログ・ファイル, 188
オペレータ・ログ・ファイル, 181
(参照 OPCOM)
creating new, 194
アクセス, 193

印刷, 194
オペレータ応答の記録, 190
管理, 196
機密保護アラーム・メッセージ, 192
クラスの使用可能化と使用禁止, 194
現在のクローズと新しいオープン, 188
再起動, 196
サイズ, 193
システム・パラメータに対する変更内容の記録, 191
省略時の状態の指定, 195
初期化メッセージ, 188
設定, 187, 193
ターミナルが使用可能または使用不能になったことを
示すメッセージ, 193
デバイス状態メッセージ, 188, 193
ページ, 196
プリント, 196
ボリュームがマウントまたはディスマウントされたこ
とを示すメッセージ, 193
ユーザからの要求とオペレータの応答を示すメッセ
ージ, 193
ユーザ要求の記録, 190
要求識別番号, 190

か

カーネル・モード
動作中のイメージの呼び出し, 118, 121
論理名, 122
会計グループ
設定, 221
外部への LAT 接続, 307, 316
会話型ブート
システム・パラメータの表示, 57
システム・パラメータの変更, 57
会話型ユーザ
性能管理のための制限, 103
カウンタ
LAT ノードの状態, 317
書き込み可能イメージ, 118
拡張 (参照 ファイル拡張)
仮想デバイス
システム・スタートアップ時のマウント, 303
監査機構
機密保護
SHOW AUDIT コマンドを使用した表示, 199
監査分析ユーティリティ, 181
(参照 ANALYZE/AUDIT コマンド)
(参照 Monitor ユーティリティ)
機密保護レポートの作成, 201
監査ログ・ファイル, 181
(参照 機密保護監査ログ・ファイル)
監視
マルチプロセッシング環境の, 329
完全な名前
DECnet-Plus
構文, 251
定義, 250

き

キーパッドの定義, 228, 229

キー・プロセス

定義, 74

記憶位置

システム・ダンプ・ファイルの, 62

システム・ファイルの

論理名による再定義, 107

スワップ・ファイル, 66

代替記憶位置の指定, 93

ページ・ファイル, 64, 66

代替ページ・ファイルの指定, 93

既知イメージ

インストール, 122

システム・スタートアップ時, 117

インストールの利点の評価, 118, 123

削除, 124

定義, 117

特権拡張, 120

表示, 123

ファイル指定, 122

ボリュームのディスマウント, 124

既知ファイル・リスト

定義, 117

機密保護, 181

OpenVMS Cluster, 237

アラーム

使用可能にする, 200

アラーム・イベントの指定, 192

アラーム・メッセージ, 192

オペレータ・ターミナルの使用可能, 192

監査

オペレータ・ターミナルを使用可能にする, 200

説明, 197

システム・ダンプ・ファイルの保護, 65

特権付きのイメージをインストールすることによって発生するリスク, 121

機密保護監査

クラスタ内の各ノード用のアーカイブ・ファイル, 198

機密保護監査機構

SHOW AUDIT コマンドを使用した表示, 199

イベントの使用禁止, 200

オペレータ・ターミナルを使用可能にする, 200

説明, 197

機密保護監査ログ・ファイル

新しいバージョンの作成, 201, 202

クローズ, 201

検討方針, 198

定義,

キャッシュ

Virtual I/O, 140

データ, 132

メタデータ, 132

キャッシング

禁止, 132

先読みキャッシングの禁止, 138

属性, 137

キャッシング属性

設定, 137

定義, 137

表示, 137

キャッシングの禁止

クラスタ全体, 132

先読み, 138

ボリューム, 133

キュー

OpenVMS Cluster 環境, 224

バッチ

ベクタ処理のための設定, 333

キュー・データベース

システム・ディスクの入出力を軽減するための移動, 107

共用アドレス・イメージ

共用既知イメージ・セクション, 119

共用アドレス・データ

Alpha システムでの使用, 119

Install ユーティリティがアドレス・データ・セクションの内容を判断, 120

P1 空間の割り当て, 120

アドレス・データ・セクション, 119

イメージ・セクション, 119

実行可能なイメージが使用できる, 120

共用可能イメージ, 118, 120

論理名の割り当て, 124

共用資源

定義, 224

禁止

メッセージの表示, 33

く

クォータ

UETP を実行するのに必要な, 166

クォータ・ファイル

ANALYZE/DISK_STRUCTURE ユーティリティがコピーを作成する, 370

QUOTA.SYS, 369

予約ファイル, 369

クラス

使用可能化と使用禁止, 194

クラッシュ・ダンプ, 62

解放, 87

システム・スタートアップ時, 85

選択型ダンプ, 64

物理ダンプ, 64

物理ダンプと選択型ダンプとの比較, 64

分析, 82

ページ・ファイルからの解放, 86

保存のための必要条件, 64

リブート時のページ・ファイルの内容の保存, 63

クラッシュ・ログ・ユーティリティ・エキストラクタ (CLUE)

説明, 83

グループ

会計, 221

グループ番号

変更, 238

グローバル・セクション, 120

グローバル・ページ, 120

クロス・アーキテクチャ
ブート, 279

け

継続ファイル

CONTIN.SYS, 369
拡張ファイル識別子として使用する, 369
予約ファイル, 369

現在の会計情報ファイル

移動, 219
更新, 219
省略時のファイル名, 219
調査対象の資源の制御, 219
調査対象の資源の調査, 218
定義, 217

現在のシステム・パラメータ, 25, 47

検索リスト

インストール済みイメージの優先順位, 117

こ

コア・イメージ・ファイル

CORIMG.SYS, 369
OpenVMS がサポートしない, 369

構成

LAN, 255
LAN の表示, 263

コマンド・ファイル

LANCP (LAN 制御プログラム) ユーティリティ, 262

コマンド・プロシージャ

AUTOGEN を定期的に行うための, 44
LIBDECOMP.COM, 108
MONITOR, 210, 213
SHOW CLUSTER, 236
SHOW_CLUSTER\$INIT.COM, 235
出力の制御, 228
初期化, 235
レポートの書式化, 231
VVIEF\$INSTAL.COM プロシージャ, 331
オペレータ・ログ・ファイルの省略時の状態の指定,
195

コントローラ構成の問題, 300

コンパクト・ディスク・ドライブ

UETP によるサポート, 151

さ

サービス

書き込み保護, 302
ノード, 307
バインド, 302
パスワード保護, 302

作業負荷

監視, 102
使用するツール, 102
管理の方針, 103
システム・パラメータの調整, 105
把握, 102
配分, 103
効率的なアプリケーションの設計, 104
ロゲインする時間帯の制限, 103

先読みキャッシング, 138

索引ファイル

INDEXF.SYS, 366
説明, 366
代替ファイル・ヘッダ, 367
内容, 366
バックアップ索引ファイル・ヘッダ, 367
バックアップ・ホーム・ブロック, 366
ビットマップ, 367
ファイル番号, 365
ファイル・ヘッダ, 367, 368
ブート・イメージ, 367
ブート・ブロック, 366, 367
ホーム・ブロック, 366, 367
予約ファイル, 366

削除

ページ・ファイル, スワップ・ファイル, ダンプ・
ファイル
新しいバージョン作成後の, 97
警告, 90

作成

オペレータ・ログ・ファイル, 194
システム・パラメータ・ファイル
SYSGEN による, 55
ページ, スワップ, ダンプ・ファイル
SYSGEN による, 96

サテライト

LAN MOP ダウンライン・ロード・サービスからの移
行, 277
LAN MOP と DECnet MOP のダウンライン・ロード・
サービス, 276
クロス・アーキテクチャ・ブート, 279
ブート, 276

サブプロセス

作成
LANCP SPAWN コマンドを使用した, 262

サポート担当者

システム障害の連絡, 82

し

資源

会計情報の収集, 217
共用, 224
使用状況の調査, 217
使用状況レポートの作成, 220

資源の会計情報の収集, 217

資源の使用状況

レポートの生成, 220

時刻, 237

時差係数 (TDF)

表, 371

システム

InfoServer によるダウンライン・ローディング, 290

システム管理

LANACP ユーティリティの終了, 260
LAN デバイス構成の表示, 263
LAN デバイス・パラメータの表示, 263
MOP ダウンライン・ロード・サービスの許可, 279
MOP ダウンライン・ロード・サービスの禁止, 279

- OPCOM メッセージの表示, 281
- カウンタのクリア, 281
- 作業
 - クラスタ全体の管理, 225
 - 性能の管理, 101
 - ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの管理, 61
- 状態とカウンタの表示, 279, 280
- デバイス情報の管理, 272
- デバイス情報の削除, 273
- ノード情報の削除, 276
- ノード情報の表示, 274
- ロード・トレース機能, 281
- システム管理ユーティリティ (SYSMAN)
 - AUTOGEN で使用するためのパラメータの変換, 27
 - OpenVMS Cluster の管理, 237
 - PARAMETERS コマンド, 47
 - システム・パラメータの管理, 26, 47
 - システム・パラメータの表示, 48
 - システム・パラメータの変更
 - アクティブ値, 49
 - システム・パラメータ・ファイルの変更, 49
 - スタートアップの制限事項, 301
- システム・シャットダウン
 - システム停止のタイミングについての注意, 62
- システム障害, 62
 - 原因の究明, 62
 - 原因の決定, 82
 - サポート担当者への連絡, 82
 - システム・ダンプ・ファイルの書き込み, 62
 - 発生後のシステム・ダンプ・ファイルの内容の保存, 85
- システム・スタートアップ
 - イメージのインストール, 117
 - 遠隔 InfoServer デバイスを使用可能にする, 303
 - システム・ダンプ・ファイルの内容の保存, 85
 - ページ・ファイルからのダンプ情報の解放, 65
 - ページ・ファイルとスワップ・ファイルのインストール, 66, 88, 89
- システム生成ユーティリティ (SYSGEN)
 - AUTOGEN で使用するためのパラメータの変換, 27
 - CREATE コマンド, 96
 - INSTALL コマンド, 88
 - SYPAGSWPFILES.COM の, 89
 - オペレータ・ログ・メッセージ, 191
 - システム・パラメータの管理, 26
 - システム・パラメータの表示, 52
 - システム・パラメータの変更, 54, 191
 - 新規システム・パラメータ・ファイルの作成, 55
 - ページ・ファイル, スワップ・ファイル, ダンプ・ファイル
 - インストール, 89
 - サイズの変更, 96
 - ページ・ファイル, スワップ・ファイル, ダンプ・ファイルの作成, 96
- システム・ダンプ
 - 選択型, 74
 - ダンプ・オフ・システム・ディスク (DOSD), 74, 76
 - プロセスの順序, 74
- システム・ダンプ・アナライザ・ユーティリティ (SDA)
 - COPY コマンド, 85
 - システム障害の原因の決定, 82
 - システム・ダンプ・ファイルの内容の保存, 85
 - システム・ダンプ・ファイルの分析, 62
 - システム・スタートアップ時, 65, 82
 - ページ・ファイルからのダンプ情報の解放, 86
- システム・ダンプ・ファイル
 - BACKUP によるコピー, 65, 86
 - UIC 機密保護による保護, 65
 - 新しいバージョン作成後の削除, 97
 - インストール, 91
 - 格納場所の制御, 92
 - 欠如, 62
 - コピー, 85
 - 最小情報の保存, 72
 - サイズ
 - AUTOGEN による計算を表示, 91
 - AUTOGEN による変更, 72
 - 机上での計算, 67
 - 計算, 63
 - 最小化, 72
 - 制御, 92
 - 操作, 92
 - 必要な, 64
 - 複数のファイルの合計を指定, 93
 - ページ・ファイルに必要な, 64
 - 変更, 90, 94, 96
 - サイズの計算, 67
 - 作成後のリブート, 97
 - 作成のための必要条件, 64
 - システム障害の原因の調査, 82
 - システム・ディスク以外のディスクに格納, 63
 - 省略時の記憶位置, 62
 - 定義, 62
 - 必要条件
 - 記憶位置, 64
 - サイズ, 64
 - 不十分なディスク空間, 72
 - 物理ダンプと選択型ダンプ, 64
 - 物理ダンプと選択型ダンプの内容比較, 72
 - 分析, 82
 - ページ・ファイルの解放, 65
 - ページ・ファイルの使用, 63
 - 保護, 88
 - メモリの選択した部分の保存, 72
 - リブート時の内容の保存, 85
- システム・ディスク
 - UETP 中のテスト・エラー, 167, 168
 - UETP テスト・イメージ, 176
 - UETP によるテスト, 175
 - UETP を実行するのに必要なディスク領域, 149
 - 空間の節約, 66
 - システムの性能向上のためのファイルの移動, 107
 - フラグメンテーション, 97
 - ページ・ファイルとスワップ・ファイルの削除, 66
- システム・ディスクの入出力の軽減, 107
- システムのカスタマイズ
 - InfoServer Client for OpenVMS の起動, 298

- 遠隔 InfoServer デバイスを使用可能にする, 303
- システムの作業負荷の配分, 103
- システムの初期化
 - マルチプロセッシング・システム, 328
- システムの停止
 - システム・ダンプ・ファイル書き込みまでの待機, 62
- システム・パラメータ
 - ALPHAVMSSYS.PAR ファイル (Alpha), 25
 - AUTOGEN 計算結果が影響を与える, 32
 - AUTOGEN で使用するための変更内容の保存, 27
 - AUTOGEN による自動設定, 25
 - DUMPTYPE, 64
 - DUMPSTYLE, 64, 72
 - ERLBUFFERPAGES, 67
 - ERRORLOGBUFFERS, 67
 - GBLPAGES, 120
 - GBLSECTIONS, 120
 - MULTIPROCESSING, 329
 - NPAGEDYN, 333
 - QUANTUM, 333
 - RMS_EXTEND_SIZE, 106
 - SAVEDUMP, 63, 64
 - SCSNODE, 298
 - SMP_CPUS, 329, 331
 - VAXVMSSYS.PAR ファイル (VAX), 25
 - VBN_CACHE_S, 132
 - VCC_FLAGS, 132
 - VCC_MAX_CACHE, 135
 - VCC_MAX_IO_SIZE, 137
 - VCC_MAXSIZE, 141
 - VCC_PTES, 142
 - VCC_READAHEAD, 138
 - VECTOR_MARGIN, 333
 - VECTOR_PROC, 331
 - WSMAX, 333
 - アクティブ値, 25, 47
 - 管理, 23
 - 機能別カテゴリ, 25
 - 現在値, 25, 47
 - 省略時の値, 25
 - 新規パラメータ・ファイルの作成
 - SYSGEN による, 55
 - 制御
 - ADD_ 接頭辞による, 40
 - MAX_ 接頭辞による, 41
 - MIN_ 接頭辞による, 41
 - MODPARAMS.DAT ファイル, 27, 39, 40
 - 値を大きくする方法, 40
 - 最小値の指定, 41
 - 最大値の指定, 41
 - 絶対値の指定, 40
 - 対称型マルチプロセッシング, 329
 - ダイナミック, 25
 - タイプ, 24
 - ダイナミック, 24
 - 汎用, 24
 - メジャー, 24
 - 定義, 24
 - ディスク上, 25
 - ALPHAVMSSYS.PAR ファイル (Alpha), 57
 - VAXVMSSYS.PAR ファイル (VAX), 57
 - 表示
 - SYSGEN による, 52
 - SYSMAN による, 48
 - 会話型ブートでの, 57
 - 標準的な変更方法, 26
 - ファイル拡張, 106
 - ブート時の初期化, 57
 - ベクタ処理, 331
 - 変更
 - AUTOGEN による, 39
 - AUTOGEN の設定のチェック, 32
 - MODPARAMS.DAT ファイルでの値の指定, 27
 - MODPARAMS.DAT ファイルの編集, 39
 - SYSGEN による, 54
 - SYSMAN による, 49
 - 会話型ブートでの, 57
 - 推奨される方法, 39
 - 標準的な方法, 26
 - 変更するための推奨方法, 27
 - 他のパラメータへの影響, 26
 - ユーザ定義可能な, 25
 - システム・パラメータの管理
 - 作業, 23
 - システム・パラメータの表示
 - SYSGEN による, 52
 - SYSMAN による, 48
 - システム・パラメータの変更
 - AUTOGEN による, 39
 - SYSGEN による, 54
 - SYSMAN による, 49
 - 推奨される方法, 39
 - 標準的な方法, 26
 - システム・ハングアップ, 164, 172
 - システム・ファイル
 - 性能向上のためのシステム・ディスクからの移動, 107
 - システム・ライブラリ
 - LIBDECOMP.COM が認識できるライブラリのリスト, 111
 - LIBDECOMP.COM コマンド・プロシージャ, 109
 - 縮小, 114
 - 展開, 108, 113
 - 復元, 106
 - システム・ライブラリの展開, 108
 - システム・ライブラリの復元 (参照 システム・ライブラリの展開)
 - 実行可能イメージ, 117, 121, 122
 - 自動構成
 - DECnet-Plus アドレス, 250
 - 常駐イメージ
 - インストール (Alpha), 118
 - 初期化ファイル
 - SHOW CLUSTER
 - 作成, 235
 - SHOW_CLUSTER\$INIT, 235, 236
 - SHOW CLUSTER レポートの設定, 228
 - 作成, 236
 - 初期システム・ロード (参照 ISL)

シリンダ
定義, 364
診断
UETP との関係, 160
シンボル
MODPARAMS.DAT ファイルで定義, 40
NUM_ETHERADAPT, 42
NUM_NODES, 42
PAGEFILEn_NAME, 92
SWAPFILEn_NAME, 92
システム・パラメータ用, 40
定義
MODPARAMS.DAT ファイル, 91
ページ・ファイル, スワップ・ファイル, ダンプ・
ファイルのサイズ, 93

す

スーパーバイザ・モード
論理名, 122
スカラ
定義, 330
スタートアップの制限事項
InfoServer Client for OpenVMS ソフトウェア, 301
PATHWORKS, 301
RSM, 301
SYSMAN, 301
ストレージ制御ブロック, 363
ストレージ制御ブロック (SCB)
ストレージ・ビットマップ・ファイル内の, 368
ストレージ・ビットマップ・ファイル
BITMAP.SYS, 368
ストレージ制御ブロック (SCB), 368
定義, 368
予約ファイル, 368
スワッピング
物理メモリとディスク上のファイルの間で情報を移動
するための, 65
スワップ・ファイル
1 次, 66
2 次, 66, 89
システム性能の向上のための作成, 107
新しいバージョン作成後の削除, 97
インストール, 91
2 次, 66
SYPAGSWPFILES.COM プロシージャによる, 89
SYSGEN による, 88
システム・スタートアップ時, 66, 88, 89
格納場所とサイズの制御, 92
管理作業, 61
記憶位置, 66
指定, 93
サイズ
AUTOGEN による計算を表示, 91
計算, 66, 71
現在, 93
指定, 92, 93
増加, 71
変更, 90, 94, 96
作成

AUTOGEN による, 93
SYSGEN による, 96
作成後のリブート, 97
サテライト上, 89
システム・スタートアップ時のディスクのマウント,
89
使用状況の監視, 66
使用量の監視, 71
性能向上のための移動, 107
定義, 65
パーシ, 97
表示, 66
フラグメンテーション, 97
フラグメンテーションが著しいことを示すメッセー
ジ, 97

せ

性能
CPU と I/O 時間の減少, 119
改善
RMS ファイル拡張パラメータの設定, 106
圧縮されたシステム・ライブラリの復元, 106
オプション, 105
会話型ユーザの数の制限, 103
効率的なアプリケーションの設計, 104
システム・ディスクの入出力の軽減, 107
システムのチューニング, 104
ハイウォータ・マークの無効化, 106
バッチ・キューによるベクタ処理性能の改善, 333
バッチ処理の導入, 103
頻繁に使用するイメージのインストール, 106, 116
ユーザがログインする時間帯の制限, 103
監視, 102
使用するツール, 102
管理
MONITOR を使用した, 211
イメージのインストール, 116
作業, 101
作業負荷管理方針, 103
作業負荷の把握, 102
作業負荷の配分, 103
システム・チューニング, 104
チューニング結果の評価, 105
ハードウェアの能力, 105
ベクタ処理による, 333
基本概念, 101
向上
AUTOGEN フィードバックによる, 32
共用アドレス・データ, 119
システム・ディスクからページ・ファイルとスワッ
プ・ファイルを削除することによる, 66
メモリの操作, 107
十分なハードウェア能力の重要性, 105
スワップ・ファイルのサイズの重要性, 71
ファイル拡張の影響, 106
ページ・ファイルのサイズの重要性, 69
ベクタ処理, 330, 333
性能管理 (参照 性能)
オプション, 105

性能の管理

- MONITOR を使用した, 211
- イメージのインストール, 116
- 作業, 101
- 作業負荷管理方針の決定, 103
- 作業負荷の把握, 102
- 作業負荷の配分, 103
- システム・チューニング, 104
- チューニング結果の評価, 105
- ハードウェアの能力の検討, 105

製品データベース

- 定義, 393

セキュリティ・オペレータ・ターミナル, 200

セクション

- グローバル, 120
- ページ, 120

セクタ

- Files-11
- 定義, 364

潜在的ベクタ使用プロセス

- 検出, 333

選択型システム・ダンプ, 64

- サイズを小さくするために使用, 72
- 順序の情報が書き込まれる, 73
- 物理システム・ダンプとの比較, 64, 72
- 保存, 72

た

ターミナル

- UETP 出力, 175
- UETP テスト・イメージ, 176
- UETP によるテスト, 173, 175
- UETP のための準備, 146, 149, 151, 158
- 機密保護アラームの, 200

ターミナル・サーバ, 308

- OpenVMS システム上, 316
- 定義, 305

対称型ベクタ処理構成, 330

ダイナミック・システム・パラメータ, 24, 25

大容量記憶制御プロトコル, 224

ダウンライン・ロード, 290

ダンプ・オフ・システム・ディスク (DOSD), 76

ダンプ・ファイル

- 上書き, 62
- 管理作業, 61
- 記憶位置
- 指定, 93
- サイズ
- 指定, 93
- 変更, 90, 94
- 作成
- AUTOGEN による, 93
- 定義, 62
- 取り込まれる情報, 62

ダンプ・ファイル情報

- 自動保存, 82

ち

チューニング, 101

(参照 性能)

- AUTOGEN による, 27
- AUTOGEN フィードバックによる最小化, 32
- 結果の評価, 105
- 定義, 104
- ハードウェアの能力の検討, 105
- 必要な時期の予測, 105

て

ディスク, 145, 363 (参照 システム・ディスク) (参照 ディスク構造) (参照 ユーザ・ディスク) (参照 ディスク空間)

Files-11

- ディレクトリ階層, 365

エクステント, 363

概念, 363

クラスタ

- 割り当ての単位, 363

構造

- 物理的, 364
- 論理的, 363

使用量

- クォータ・ファイルに保存される UIC, 369

シリンダ

- 定義, 364

トラック

- 定義, 364

入出力性能, 107

ファイル識別子, 365

フラグメンテーション, 97

ブロック

- クラスタ別分類, 363

ディスク I/O リソース

オフロード

- Virtual I/O Cache, 140

ディスク記憶サーバ, 298

ディスク空間

クラスタ別割り当て, 363

使用量の調査, 222

節約

- 選択型ダンプによる, 64
- ページ・ファイルとスワップ・ファイルをシステム・ディスクから削除することによる, 66

保存

- 最小ダンプ情報を格納することによる, 72

ディスク構造

Files-11, 364

ディスク・ボリューム

初期化, 363

定義, 363

マウント

- システム・スタートアップ時, 303
- ページ・ファイルとスワップ・ファイルのための, 89

ディスク領域

未使用領域についての決定, 108

ディスクマウント

- DECdtm トランザクション・ログを持ったディスク, 352

- データ・クラス
 - SHOW CLUSTER, 227
 - 削除, 233
- データ・フィールド
 - SHOW CLUSTER, 227
 - 削除, 233
- データベース
 - LAT データベース, 316
- テープ
 - UETP テスト・イメージ, 176
 - UETP によるテスト, 173, 175
 - UETP のための準備, 146, 148, 150
- テープ・カートリッジ・ドライブ
 - UETP のための準備, 150
- テープ大容量記憶制御プロトコル, 224
- テスト
 - UETP 中のエラー, 169
- テストのロード
 - 個々の実行, 157
- デバイス
 - LANCP (LAN 制御プログラム) ユーティリティによる管理, 262
 - イーサネット・アダプタ
 - AUTOGEN で使用する数の指定, 42
 - 状態レポート, 188
- デバイス・テスト
 - UETP による個々の実行, 174
- デュアル・アーキテクチャ OpenVMS Cluster システム
 - イメージのインストール, 239
 - 例, 239

と

- 動的負荷分散, 328
- 特権
 - UETP に必要な, 166
 - インストール・ファイルのための拡張, 120
 - 必要な特権を持たないユーザによるプログラム実行を可能にする, 121
- 特権イメージ, 118, 121
- トラック
 - 定義, 364
- トラブル・シューティング
 - UETP, 164
 - システム障害, 82
 - システム・ダンプ・ファイル, 82
 - システムのハング, 86
- トランザクション
 - DECdtm
 - 監視, 344
- トランザクション・グループ
 - 定義, 341
 - 例, 341
- トランスポート, 294
 - LASTport, 293
 - ローカル・エリア (LAT), 305
- トリガ・ブート
 - MOP ダウンライン・ロード・サービス, 282

な
長いレポート形式 (参照 UETP 中のコンソール・レポート)

に
入出力

- システム・ディスクの軽減, 107

ね
ネームスペース

- DECnet-Plus の完全な名前, 251

ネットワーク

- Compaq のネットワーク・オプション, 244
- LAN の機能拡張, 255
- 可能なソフトウェアの組み合わせ, 245
- ネットワーク・ソフトウェア・オプション, 244

ネットワーク管理

- 概念, 247, 251
- 分散, 247, 251

ネットワーク制御言語, 250

ネットワーク制御プログラム管理

- Phase IV ノードの, 251

の
ノード

- LAT データベース内, 316
- ノード情報の削除, 276
- ノード情報の設定, 275

ノード名

- DECnet Phase IV, 250
- DECnet-Plus の完全な名前, 250
- OpenVMS InfoServer Client のスタートアップ時に必要な, 298
- 命名規則, 250

は
パーティション, 290

ハードウェア

- 十分なハードウェア能力の重要性, 105

ハイウォータ・マーク

- システムの性能改善のための無効化, 106

バグ・チェック

- UETP 中のメッセージ, 172

バケット, 247

パスワード

- クラスタにアクセスするための変更, 238

バックアップ・ユーティリティ (BACKUP)

- セーブ・セット・ファイルを MFD に入れる, 369
- ダンプ・ファイルのコピー, 65

バックアップ・ログ・ファイル

- BACKUP.SYS, 369
- 予約ファイル, 369

バッチ・キュー

- ベクタ処理のための設定, 333

バッチ・ジョブ

- システムの作業負荷の配分, 103

バッファ

- LAT 環境での大容量のバッファの使用, 315
- パラメータ・ファイル
 - ALPHAVMSSYS.PAR (Alpha), 25
 - ブート時のパラメータの初期化, 57
 - AUTOGEN による複数の使用, 43
 - MODPARAMS.DAT, 38
 - 例, 39
 - MODPARAMS.DAT ファイルへの取り込み, 43
 - OpenVMS Cluster 環境で共通な, 43
 - VAXVMSSYS.PAR (VAX), 25
 - ブート時のアクティブ・パラメータの初期化, 57
 - エラー・チェックに関する制限, 40
 - 省略時の値, 25
 - 新規作成
 - SYSGEN による, 55
 - 変更
 - SYSGEN による, 54
 - SYSMAN による, 49
 - 影響, 54

ひ

- 非対称型ベクタ処理構成, 330

ふ

- ファイル

- AUTOGEN フィードバック, 32
- システム
 - システム・ディスクの入出力軽減のための移動, 107
- 命名
 - Files-11 ボリューム, 367
 - 予約, 365
- ファイル・アクセス
 - 同時に行われる数の表示, 106
- ファイル拡張
 - サイズの指定, 106
 - システムの性能への影響, 106
 - 制御するシステム・パラメータ, 106
- ファイル・シーケンス番号, 363
 - (参照 ファイル識別子)
- ファイル識別子, 363
 - Files-11, 365
 - 相対ボリューム番号 (RVN), 365
 - ファイル・シーケンス番号 (SEQ), 365
 - ファイル番号, 365
- ファイル指定
 - Files-11 ボリューム, 367
 - イメージをインストールする, 122
- ファイルのコピー
 - ダンプ・ファイル, 86
- ファイルのフラグメンテーション
 - ページ・ファイルとスワップ・ファイルの, 97
- ファイル・ヘッダ
 - 索引ファイル, 367
 - 1 次, 368
 - 拡張, 368
 - 内容, 368
- ファイル保護
 - SYSDUMP.DMP ファイル, 65
- フィックスアップ

- インストール時の実行, 119
- ブート
 - VVIEF コードのロード, 331
 - クロス・アーキテクチャ, 279
 - ブートストラップ・イメージ
 - 索引ファイル, 367
 - ページ・ファイルとスワップ・ファイルのインストール, 66
 - マルチプロセッシング・システム, 328
- ブート・ブロック
 - 索引ファイル, 367
- 付加されたプロセッサ, 328
- 負荷分散
 - LAT ソフトウェアの使用, 306
- 物理システム・ダンプ, 64
 - 選択型システム・ダンプとの比較, 64, 72
- 物理メモリ
 - 共有アドレス・データを使用して節約, 119
- 不良ブロック・ファイル
 - BADBLK.SYS, 369
 - 説明, 369
 - 予約ファイル, 369
- ブレイクイン
 - 監査機構の試み, 197
- ブレイクインの試み
 - 監査, 197
- プロシージャ実行 (@) コマンド, 237
- プロセス・クォータ
 - ベクタ処理のための調整, 333
- プロセス・ダンプ
 - 機密保護, 98, 99
 - ディスク制限, 98
 - 特権ユーザ, 98
- プロセスの制限
 - ベクタ処理のための調整, 333
- プロセッサ, 329
 - マルチプロセッシングのアクティブ・セットからの削除, 329
 - マルチプロセッシングのアクティブ・セットへの追加, 329
- ブロック
 - Files-11
 - 定義, 363
- プロトコル
 - LASTport, 293
 - LASTport/Disk, 293
 - LASTport/Tape, 294
- 分散ネットワーク管理
 - 概念, 247, 251

へ

- ページ・ファイル
 - 1 次, 66
 - 2 次, 66, 89
 - システム性能の向上のための作成, 107
 - 新しいバージョン作成後の削除, 97
 - インストール, 91
 - 2 次, 66
 - SYSPAGSWPFILES.COM プロシージャによる, 89

SYSGEN による, 88
 システム・スタートアップ時, 66, 88, 89
 解放, 87
 格納場所とサイズの制御, 92
 管理作業, 61
 記憶位置
 指定, 93
 サイズ
 AUTOGEN による計算を表示, 91
 計算, 66, 68, 69
 現在, 93
 サイズを増やす場合, 70
 システム・ダンプの保存に必要, 68
 指定, 92, 93
 必要, 69
 変更, 90, 94, 96
 サイズの変更
 推奨方法, 85
 作成
 AUTOGEN による, 93
 SYSGEN による, 96
 作成後のレポート, 97
 サテライト上, 89
 システム・スタートアップ時のディスクのマウント,
 89
 システム・ダンプ・ファイルとして, 63, 67, 88
 ダンプの解放, 86
 必要なサイズ, 64
 使用状況の監視, 66
 使用量の制限, 71
 性能向上のための移動, 107
 ダンプ情報の解放, 65, 86
 ダンプの解放, 87
 定義, 65
 ページ, 97
 必要条件
 記憶位置, 64, 66
 ダンプを保存するためのサイズ, 64
 表示, 66
 フラグメンテーション, 97
 メッセージ
 サイズが不十分であることを示す, 70
 フラグメンテーションが著しいことを示す, 97
 要件
 記憶位置, 107
 レポート時のダンプ内容の保存, 63
 ページ・ファイル, スワップ・ファイル, ダンプ・ファ
 イルの管理
 作業, 61
 ページ・ファイル, スワップ・ファイル, ダンプ・ファ
 イルの作成
 AUTOGEN による, 93
 ページ・ファイルとスワップ・ファイルのインストール
 AUTOGEN による, 91
 SYPAGSWPFILES.COM コマンド・プロシージャによ
 る, 88
 SYSGEN による, 88
 システム・スタートアップ時, 66, 88, 89
 ページ・ファイルのサイズ
 ダンプを保存するために必要な, 64
 ページング, 65
 ベクタ処理による増加, 333
 ベクタ
 定義, 330
 ベクタ CPU 時間
 プロセスに関する情報の入手, 335
 プロセスの情報の入手, 334
 ベクタ機能, 330
 ACL の設定, 333
 システム内で利用可能かどうかの判断, 335
 ベクタ機能付きシステム, 330
 ベクタ機能付きプロセッサ
 識別, 335
 システム構成からの削除, 331
 システム構成への追加, 331
 利用できない場合, 332
 ベクタ・コンテキスト・スイッチ
 についての情報の入手,
 ベクタ使用プロセス
 管理, 332
 潜在的, 333
 についての情報の入手, 334
 判定, 334
 ベクタ処理, 330, 335
 VAX サポート, 330
 管理, 331
 管理の作業, 331
 資源の要件, 333
 システム性能, 330
 システムの構成, 331
 システムのチューニング, 333
 定義, 330
 についての情報の入手, 334
 のためのバッチ・キューの設定, 333
 ベクタ・プロセッサ数の入手, 335
 ベクタ処理環境の管理
 作業, 331
 ベクタ・レジスタ, 330
 ヘッダ常駐イメージ, 117, 119

ほ
 ホーム・ブロック
 索引ファイル, 367
 保護
 ファイル
 システム・ダンプ・ファイル, 65
 保護イメージ, 118, 121, 122
 保守操作プロトコル (参照 MOP (保守操作プロトコル))
 保留不良ブロック・ログ・ファイル
 BADLOG.SYS, 369
 予約ファイル, 369
 ボリューム
 ディスマウント
 制限, 124
 ボリューム機密保護プロファイル
 SECURITY.SYS, 370
 予約ファイル, 370
 ボリューム・セット

VOLSET.SYS に入る情報, 369
ボリューム・セット・リスト・ファイル
ANALYZE/DISK_STRUCTURE ユーティリティが使用する, 369
VOLSET.SYS, 369
説明, 369
予約ファイル, 369
ボリュームのマウント
キャッシングを禁止した状態, 133
ディスク
システム・スタートアップ時, 303

ま

マスタ・コマンド・プロシージャ (参照 UETP.COM プロシージャ)
マスタ・ファイル・ディレクトリ, 363, 365
BACKUP がセーブ・セット・ファイルを入れる, 369
説明, 369
予約ファイル, 369
マルチプロセッシング, 328
監視, 329
システム・パラメータ, 329
情報の表示, 330
定義, 328
マルチプロセッシング環境の管理, 329
作業, 329

み

短いレポート形式 (参照 UETP 中のコンソール・レポート)

め

メール・ユーティリティ (MAIL)
による AUTOGEN レポートの送信, 44
メッセージ
エラー
削除, 231
オペレータの応答, 190
機密保護アラーム, 192
使用開始と使用停止, 189
表示の禁止, 33
ブロードキャスト
削除, 232
ページ・ファイルのサイズが不十分であることを示す, 70
ページ・ファイルまたはスワップ・ファイルのフラグメンテーションが著しいことを示す, 97
ベクタ・プロセッサが利用できないことを示す, 332
ユーザ要求, 190
メモリ
イメージ, 118
イメージのインストールによる効率的な使用, 107
大きすぎて完全なシステム・ダンプを保存できない場合, 72
キャッシュ, 131
共用可能イメージによる節約, 120
クラッシュ・ダンプに取り込まれる情報, 62
選択型ダンプ, 64, 72
物理ダンプ, 64, 72

スワッピング, 65
セクション, 120
ページング, 65
ベクタ使用プロセスへの割り当て, 333
メモリ常駐セクション, 125

や

矢印キー
各キーの機能, 229

ゆ

ユーザ
会話型ユーザ数の制限, 103
ログインする時間帯の制限, 103
ユーザ環境テスト・パッケージ (参照 UETP(ユーザ環境テスト・パッケージ))
ユーザ・ディスク
UETP 中のテスト・エラー, 167
UETP テスト・イメージ, 176
UETP によるテスト, 175
UETP のための準備, 146, 149, 150
UETP のための領域の要件, 149
ユーザ・ファイル・ディレクトリ, 363
ユーザ負荷
UETP DECnet テストのために定義された, 178
UETP ロード・テストの決定に使用される公式, 162
UETP ロード・テストのための定義, 158
ユーザ・モード
論理名, 122

よ

要求識別番号
オペレータ・ログ・ファイルに示される, 190
読み書き可能ディスク
パーティション, 290
予約ファイル, 365
クォータ・ファイル (QUOTA.SYS), 369
継続ファイル (CONTIN.SYS), 369
索引ファイル (INDEXF.SYS), 366
ストレージ・ビットマップ・ファイル (BITMAP.SYS), 368
バックアップ・ログ・ファイル (BACKUP.SYS), 369
不良ブロック・ファイル (BADBLK.SYS), 369
保留不良ブロック・ログ・ファイル (BADLOG.SYS), 369
ボリューム機密保護プロファイル (SECURITY.SYS), 370
ボリューム・セット・リスト・ファイル (VOLSET.SYS), 369
マスタ・ファイル・ディレクトリ (MFD), 369
リスト, 365
予約メモリ・レジストリ, 125
エントリの削除, 127
エントリの追加, 126
使用, 125, 129
予約メモリの解放, 127
予約メモリの表示, 128
予約メモリの復帰, 129
予約メモリの割り当て, 127

ら

ライトスルー・キャッシュ, 132

ライン

TCP/IP

ホストの接続, 247

ライン・プリンタ

UETP 出力, 175

UETP テスト・イメージ, 176

UETP によるテスト, 173

UETP のための準備, 146, 149, 151

り

リポート

システム・ファイル作成後の, 97

利用者識別コード (参照 UIC)

UETP のための, 150

リンク可能イメージ, 117

る

ルーティング

定義, 247

れ

レキシカル関数

F\$GETJPI, 334

F\$GETSYI, 334

ベクタ処理に関する情報の入手, 334

レポート

AUTOGEN, 44

SHOW CLUSTER

移動, 233

クラスの追加, 227

構成, 227

コマンド・プロシージャによる制御, 228

スクロール, 229, 235

スタート時の省略時の値の変更, 235

データの追加, 230

表示内容の制御, 228, 231, 232

フィールドの追加, 227

変更のためのコマンド, 231

レポートの圧縮, 233

ろ

ローカル・エリア・システム・トランスポート制御プログラム, 289

ローカル・エリア・ネットワーク (参照 LAN)

ローカル・エリア・ネットワーク補助制御プログラム (参照 LANACP ユーティリティ)

ローカル・ページ・ファイルとローカル・スワップ・ファイル

SATELLITE_PAGE.COM プロシージャによるインストール, 89

ロード・テスト

UETP のためのユーザ負荷の定義, 158

説明, 176

ロード・トレース機能, 281

ロード・レベリング

動的, 328

ログ・ファイル

UETP により作成された

NETSERVER.LOG, 169

OLDUETP.LOG, 164

ロード・テスト時, 170

オペレータ

管理, 196

記憶位置の指定, 193

記録するクラスの使用可能化と使用禁止, 194

再起動, 196

新規作成, 194

設定, 193

プリント, 196

監査の指定

検討方針, 198

機密保護監査,

新しいバージョンの作成, 201

システム・ディスクの入出力を軽減するための移動, 107

論理名

ACCOUNTNG, 220

AGEN\$FEEDBACK_REQ_TIME, 43

MON\$ARCHIVE, 211

OPC\$LOGFILE_CLASSES, 195

OPC\$LOGFILE_ENABLE, 195

OPC\$LOGFILE_NAME, 193, 195

OPC\$OPAO_ENABLE, 195

SHOW_CLUSTER\$INIT, 236

SYS\$AUDIT_SERVER_INHIBIT, 199

SYS\$DECDTM_INHIBIT, 358

SYS\$ERRORLOG, 182

SYS\$JOURNAL, 338

SYS\$MONITOR, 211

UETP によって使用される

CTRLNAME, 174

LOADS, 177

SYS\$INPUT, 174

SYS\$OUTPUT, 175

共用可能イメージのための割り当て, 124

システム・ファイルの記憶位置の再定義, 107

信用, 122

特権モード, 122

わ

ワーキング・セット

ベクタ化アプリケーションのための調整, 333